TP 14565E

DEVELOPMENT OF AN OPC SERVER FOR NTCIP FOR ITS EQUIPMENT

Prepared for
Transport Canada

by

Tecsult Inc.

DECEMBER 2005

Project Reference: 2004RDA-08
Tecsult Reference: 05-12757

TECSULT Matrikon

TP 14565E

DEVELOPMENT OF AN OPC SERVER FOR NTCIP FOR ITS EQUIPMENT

by

Jean-Pierre Lauzon, P.Eng.
Tecsult

with the participation of:

Éric Bertrand, Eng., Tecsult
Chris Robinson, Matrikon

DECEMBER 2005

This project is part of Canada's Intelligent Transportation Systems (ITS) R&D Plan, *Innovation Through Partnership*, funded by the ITS Office of Transport Canada under the Strategic Highway Infrastructure Program (SHIP).

The Transportation Development Centre of Transport Canada served as technical authority of this project.

This report reflects the views of the authors and not necessarily those of Transport Canada.

Transport Canada does not endorse products or manufacturers. Trade or manufacturer's names appear in this report only because they are essential to its objectives.

Un sommaire français se trouve avant la table des matières.

2007-00850

DOCUMENT:	Development of an OPC Server for NTCIP for ITS Equipment
DESCRIPTION:	Final Report
PROJECT:	0512757
DATE:	December 23rd, 2005
CODIFICATION:	0512757 – Rév. 01

PREPARED BY:	Jean-Pierre Lauzon, P.Eng.
WITH THE PARTICIPATION OF:	Éric Bertrand, Eng., Tecsult Chris Robinson, Matrikon

---

Transport Canada / Transports Canada

PUBLICATION DATA FORM

1. Transport Canada Publication No.	2. Project No.	3. Recipient's Catalogue No.
TP 14565E		

4. Title and Subtitle	5. Publication Date
Development of an OPC Server for NTCIP for ITS Equipment	December 2005
	6. Performing Organization Document No. 05-12757

7. Author(s)	8. Transport Canada File No.
Jean-Pierre Lauzon, Éric Bertrand, Chris Robinson	2450-GP746

9. Performing Organization Name and Address		10. PWGSC File No.
Tecsult 85 Ste Catherine St. West Montreal, Quebec H2X 3P4	Matrikon Inc #1800, 10405 Jasper Ave Edmonton, Alberta T5J 3N4	11. PWGSC or Transport Canada Contract No.

12. Sponsoring Agency Name and Address		13. Type of Publication and Period Covered
Transportation Development Centre (TDC) 800 René Lévesque Blvd. West Suite 600 Montreal, Quebec H3B 1X9	ITS Office – Transport Canada Place de Ville, Tower C, Floor 27 330 Sparks Street Ottawa, Ontario K1A 0N5	Final
		14. Project Officer Pierre Bolduc

15. Supplementary Notes (Funding programs, titles of related publications, etc.)

This project is part of Canada's Intelligent Transportation Systems (ITS) R&D Plan, *Innovation Through Partnership*, funded by the ITS Office of Transport Canada under the Strategic Highway Infrastructure Program (SHIP). The Transportation Development Centre of Transport Canada served as technical authority for the plan.

16. Abstract

This report summarizes the development and testing of a communication driver that permits communication between NTCIP-compliant ITS field equipment, such as dynamic message signs, and a central control facility.

Traffic management centres in large cities typically interface with a wide variety of field equipment, such as dynamic message signs, traffic light controllers, and surveillance cameras. To enable and ease the integration of equipment from different manufacturers, standards known as the National Transportation Communications for ITS Protocol (NTCIP) has been developed. In the manufacturing sector, however, commercially available systems that use Supervisory Control and Data Acquisition (SCADA) have been standardized using an interface known as OPC.

This project developed an OPC communications server designed to act as a bridge between NTCIP compliant ITS equipment and a SCADA-based traffic management system. The project involved the following tasks:

- determination of the functional requirements of the OPC driver in collaboration with Transports Québec (MTQ)
- design and development of the software application
- bench testing of the application
- integrated testing in an ITS environment, i.e., an MTQ pilot project
- evaluation of the results

The prototype successfully developed in this project is being used in traffic management systems in the U.S. and Canada.

17. Key Words	18. Distribution Statement
Traffic control centres, data communications, intelligent transportation systems	This document is available online at www.tc.gc.ca/tdc/menu.htm

19. Security Classification (of this publication)	20. Security Classification (of this page)	21. Declassification (date)	22. No. of Pages	23. Price
Unclassified	Unclassified	—	x, 12, apps	Shipping/ Handling

Transports Canada Transport Canada

FORMULE DE DONNÉES POUR PUBLICATION

1. N° de la publication de Transports Canada	2. N° de l'étude	3. N° de catalogue du destinataire
TP 14565E		

4. Titre et sous-titre	5. Date de la publication
Development of an OPC Server for NTCIP for ITS Equipment	Décembre 2005
	6. N° de document de l'organisme exécutant 05-12757
7. Auteur(s) Jean-Pierre Lauzon, Éric Bertrand, Chris Robinson	8. N° de dossier - Transports Canada 2450-GP746

9. Nom et adresse de l'organisme exécutant		10. N° de dossier - TPSGC
Tecsult 85, rue Saint-Catherine Ouest Montréal (Québec) H2X 3P4	Matrikon Inc #1800, 10405 Jasper Ave Edmonton (Alberta) T5J 3N4	11. N° de contrat - TPSGC ou Transports Canada

12. Nom et adresse de l'organisme parrain		13. Genre de publication et période visée
Centre de développement des transports (CDT) 800, boul. René-Lévesque Ouest Bureau 600 Montréal (Québec) H3B 1X9	Bureau des STI – Transports Canada Place de Ville, Tour C, 27e étage 330, rue Sparks Ottawa, Ontario K1A 0N5	Final
		14. Agent de projet Pierre Bolduc

15. Remarques additionnelles (programmes de financement, titres de publications connexes, etc.)

Ce projet fait partie du Plan de R&D du Canada sur les systèmes de transports intelligents (STI), *Innover par l'établissement de partenariats*, financé par le Bureau des STI ainsi que par Transports Canada dans le cadre du programme stratégique d'infrastructures routières (PSIR). Le Centre de développement des transports de Transports Canada a agi comme responsable technique pour ce plan.

16. Résumé

Ce rapport résume les travaux de développement et d'essai d'un pilote de communications qui permet des échanges entre des périphériques STI répondant aux normes NTCIP, comme des panneaux à message variable et un centre de régulation.

Les centres de gestion de la circulation des grandes villes dialoguent avec toutes sortes d'équipements sur le terrain, comme des panneaux à message variable, des régulateurs de feux de circulation et des caméras de surveillance. Pour faciliter l'intégration d'équipements de différents constructeurs, un ensemble de normes de communication, connues sous le nom de NTCIP (*National Transportation Communications for ITS Protocol*) ont été élaborées. Mais dans le secteur manufacturier, c'est une interface appelée OPC qui pilote les communications entre les systèmes offerts sur le marché utilisant la plate-forme SCADA (*Supervisory Control and Data Acquisition*).

Le présent projet a consisté à développer un serveur de communications OPC pouvant servir de pont entre des équipements STI répondant aux normes NTCIP et un système de gestion de la circulation utilisant SCADA. Il a comporté les tâches suivantes :

- détermination des exigences fonctionnelles du pilote OPC, en collaboration avec le ministère des Transports du Québec (MTQ)
- conception et développement du logiciel d'application
- mise au banc de l'application
- essais intégrés dans un environnement STI, c.-à-d. dans le cadre d'un projet pilote du MTQ
- évaluation des résultats

Le prototype développé au cours du projet s'est avéré une réussite et il est maintenant utilisé dans des systèmes de gestion de la circulation aux États-Unis et au Canada.

17. Mots clés	18. Diffusion
Centres de régulation de la circulation, communications de données, systèmes de transports intelligents	Ce document est disponible en ligne à www.tc.gc.ca/cdt/menu.htm

19. Classification de sécurité (de cette publication)	20. Classification de sécurité (de cette page)	21. Déclassification (date)	22. Nombre de pages	23. Prix
Non classifiée	Non classifiée	—	x, 12, ann	Port et manutention

CDT/TDC 79-005
Rev. 96

Canada

EXECUTIVE SUMMARY

Traffic management centres in large cities typically interface with a wide variety of field equipment, such as dynamic message signs, traffic light controllers, and surveillance cameras. To enable and ease the integration of equipment from different manufacturers, standards known as the National Transportation Communications for ITC Protocol (NTCIP) has been developed. In the manufacturing sector, however, commercially available systems that use Supervisory Control and Data Acquisition (SCADA) have been standardized using an interface known as OPC.

Some traffic management authorities have developed their traffic management systems on a SCADA platform. This project was intended to developed an OPC communications server designed to act as a bridge between NTCIP compliant ITS equipment and a SCADA-based traffic management system.

The functional requirements for the OPC server were determined by a technical committee composed of specialists from the Ministère des transports du Québec (MTQ) actively involved in traffic management and testing of ITS equipment. Tecsult was responsible for managing the relationship and coordinating the activities with the MTQ technical committee.

Prototype development was carried out by the Matrikon development team during the September to November 2004. A beta version of the OPC server was produced for field testing in Montreal in December 2004.

The initial field testing was carried out with fixed and mobile DMS equipment provided by the MTQ. After these initial field tests, additional development was carried out to make necessary adjustments to the beta version and to develop and integrate some missing functionalities into the server; and, to finalize the prototype.

The final version of the prototype was developed during January to March 2005. It was tested in April using equipment from the MTQ Pont Laviolette project in Trois-Rivières, Quebec. The tests were successful. Small adjustments were made after the tests. The OPC server was then successfully integrated into the project and is still active and operational to this date.

SOMMAIRE EXECUTIF

Les centres de gestion de la circulation des grandes villes dialoguent avec toutes sortes d'équipements sur le terrain, comme des panneaux à message variable, des régulateurs de feux de circulation et des caméras de surveillance. Pour faciliter l'intégration d'équipements de différents constructeurs, un ensemble de normes de communication, connues sous le nom de NTCIP (*National Transportation Communications for ITS Protocol*), ont été élaborées. Mais dans le secteur manufacturier, c'est une interface appelée OPC qui pilote les communications entre les systèmes offerts sur le marché utilisant la plate-forme SCADA (*Supervisory Control and Data Acquisition*).

Certains services de gestion de la circulation ont élaboré leurs systèmes autour d'une plate-forme SCADA. Ce projet visait à développer un serveur de communications OPC pouvant servir de pont entre des équipements STI répondant aux normes NTCIP et un système de gestion de la circulation utilisant SCADA.

Les exigences fonctionnelles du serveur OPC ont été déterminées par un comité technique composé de spécialistes du ministère des Transports du Québec (MTQ) intéressés par la gestion de la circulation et l'essai d'équipements STI. Tecsult était responsable de la gestion des rapports et de la coordination des activités avec le comité technique du MTQ.

Le développement du prototype a été réalisé par l'équipe de Matrikon de septembre à novembre 2004. Une version bêta du serveur OPC a été produite pour des essais en service réel qui ont eu lieu à Montréal en décembre 2004.

Les premiers essais ont mis en jeu des panneaux à message variable fixes et mobiles fournis par le MTQ. Les travaux de développement se sont ensuite poursuivis : des améliorations ont été apportées à la version bêta et de nouvelles fonctionnalités ont été développées et intégrées au serveur. Finalement, on a mis la touche finale au prototype.

La mise au point de la version définitive du prototype s'est faite de janvier à mars 2005. En avril, le prototype était mis à l'essai avec les panneaux mis en place par le MTQ au pont Laviolette à Trois-Rivières, au Québec. Ces essais se sont révélés concluants. Après de légères modifications, le serveur OPC a été intégré avec succès au projet et il est encore opérationnel à ce jour.

TABLE OF CONTENTS

page

1.0 INTRODUCTION 1
1.1 Overview 1
1.2 Objectives 2
1.3 References 2
1.4 List of acronyms 3
1.5 project teams 3
1.6 Acknowledgements 4
2.0 WORK DONE AND RESULTS 5
2.1 Work plan and methodology 5
2.2 Functional Requirements 5
2.3 Design and development 5
2.4 Prototype development 6
2.5 Test plans and tests 6
2.6 Results and analysis 7
2.6.1 Preliminary Site Testing – MTQ Montreal (December 2004) 7
2.6.2 Additional Development (January-March 2005) 7
2.6.3 Project Testing and Integration – Pont Laviolette (April 2005) 8
2.6.4 MTQ Final Acceptance Testing (May 2005) 8
2.6.5 TC Final Acceptance (August 2005) 9
2.7 Corrective Measures and final version 10
3.0 SUMMARY OF CURRENT TECHNOLOGICAL ISSUES 10
3.1 Problems identified or remaining 10
4.0 CONCLUSIONS AND RECOMMENDATIONS 10
4.1 Results of work done vs objectives 10
4.2 Next steps envisaged 12
4.2.1 Future Development 12
5.0 SUCCESSES AND BENEFITS 12
5.1 Integration of the OPC server in traffic management center operations 12
5.2 Presentations at conferences 12

LIST OF APPENDICES

Appendix A-1 – Project Execution Plan
Appendix A-2 – Project Work Program
Appendix B-1 – Specific Requirements
Appendix B-2 – Specific Requirements – Comments
Appendix C – Rapport d'essais sur site
Appendix D – Rapport final d'essais sur site

1.0 INTRODUCTION

1.1 OVERVIEW

Typically, a Traffic Management Control Centre must interface with various field equipment as Dynamic Message Signs (DMS), traffic light controllers, surveillance cameras, vehicle detection and classification systems, meteorological stations, etc.

In order to enable and ease the integration of the various field equipment from different manufacturers, a collection of open communication standards called NTCIP has been developed. NTCIP defines a whole series of services depending on the type of transportation media being used and the specific equipment that are being addressed by a focalized standard.

Basically, all communications use the SNMP message structure and the message can transit via serial modems, leased telephone lines, fibre optics, wireless systems, etc. Depending on the capability of the equipment manufacturer, TCP/IP can be used over different links.

The OPC Server for NTCIP is an OPC server that will be designed to gather data from a variety of various NTCIP compliant DMS devices. These devices (Dynamic Message Signs) are used by the various provincial Department of Transportation throughout metropolitan areas primarily in order to provide traffic information to passing motorists.

These signs can be both stationary and mobile and are accessed through different telecommunications systems. This traffic information is a critical element in various traffic control operations to redirect traffic or to warn motorists of potential upcoming hazards.

Communication to DMS devices is typically managed through proprietary software control applications specifically designed by each manufacturer. In a multi-devices system, this can be a real problem as traffic management operators must have knowledge as to which sign must be configured, who the manufacturer of that sign is, and which control package is used to deal with that particular sign. A unified operator interface would provide a practical solution to this operation problem.

The overall goal of this R&D project is to provide a common and open way to communicate to these DMS devices using a technology already well established within the industrial market called OPC. The intention is that once the data is represented in this generic way, any OPC enabled control system package can then be used to set-up, configure and monitor any type of NTCIP compliant DMS device.

An OPC server will further facilitate systems integration for those who build traffic management systems using a SCADA platform.

The OPC server approach is efficient and is independent of any manufacturer's proprietary software control application. It will enable traffic management control operators to use the same software control package to manage any type of DMS device, thus facilitating and making more efficient the management of these devices, enabling the customization of interfaces in ways that make the traffic management tasks easier to deal with and also reducing overall training requirements.

Hopefully, in the future, this OPC server technology can be expanded to enable the integration of other types of ITS field equipment devices and provide the same benefits and solutions to other segments within the transportation industry.

1.2 OBJECTIVES

The main objective of the project was to develop, test and implement an OPC NTCIP compliant server for DMS equipment in an effort to standardize communications between different ITS-DMS field equipment.

Basically, the server has to support a certain number of communications profiles and a set of NTCIP objects as per the specifications document and its implementation as per the NTCIP reference tester and reference DMS (two manufacturers).

Specifically, the objectives of the project were the following:

a) design and development of an OPC server for NTCIP (1203-DMS) equipment;
b) testing of the OPC server on two of the Ministry of Transportation of Quebec DMS equipment and evaluation;
c) integration the OPC server into an existing MTQ project/operations; and,
d) release of software product and documentation for commercial purpose.

1.3 REFERENCES

The following internet sites are pertinent to this project and will provide the reader with more detailed information on the respective organizations and their activities, Transport Canada and the companies involved in the project.

- www.ntcip.org
- www.opcfoundation.org
- www.its-sti.gc.ca
- www.tc.gc.ca/tdc/projects/its
- www.matrikon.com
- www.tecsult.com

1.4 LIST OF ACRONYMS

TERMS AND ABBREVIATION	DESCRIPTION
ASC	Actuated Traffic Signal Controller
CDPD	Cellular Digital Packet Data
CCTV	Closed Circuit Television
DCM	Data Collection and Monitoring Devices
DMS	Dynamic Message Sign
ESS	Environmental Sensor Stations
FAT	Factory Acceptance Test
FSK	Frequency Shift Key
IP	Internet Protocol
ITS	Intelligent Transportation Systems
LAN	Local Area Network
MIB	Management Information Base
MTQ	Ministère des Transports du Québec
NTCIP	National Transportation Communications for ITS protocol
OLE	Object Linking and Embedding
OPC	OLE for Process Control. The communication standards published by the OPC Foundation
PICS	Protocol Implementation Conformance Statement
PMPP	Point-to-Multi-Point Protocol
PPP	Point-to-Point Protocol
SAT	Site Acceptance Test
SCADA	Supervisory Control and Data Acquisition, a term used to describe the control and monitoring of remote field devices
SNMP	Simple Network Management Protocol
TC	Transport Canada
UDP	User Datagram Protocol
WAN	Wide Area Network

1.5 PROJECT TEAMS

The following personnel participated in the project in various capacities:

Ministère des Transports du Québec

- Alain Dubé, Manager Project Services
- Ivan Ruscitti, A/Manager Project Services, Montreal
- Claude Marquis, Manager, Centre de signalisation, Quebec
- Jean-Claude Dyotte, Traffic Technician, MTQ Montreal
- Bruno Mathieu, Traffic Engineer, MTQ Quebec
- Patrick Nichol,
- Michel Marmette,
- Mervat Tannous, Project Services, MTQ Montreal
- Jean Douville, MTQ Trois-Rivières (pilot project)

Transport Canada – Management Committee

- Madeleine Betts, Transport Canada – Intelligent Transport Systems (ITS) Office
- Pierre Bolduc, Transport Canada - Transport Development Center (TC/TDC)
- Richard Chylinski., Manager Software Systems, Transportation System Div., Delcan

Tecsult

- Eric Bertand, Lead Project Engineer
- Jean-Pierre Lauzon, Project Manager
- André-Marie Bernatchez, Specialist, Automation and Controls
- Michel Savard, Manager Automation and Controls

Matrikon

- Chris Robinson, OPC Development Manager
- Ryan Sadler, System Developer
- Eric Murphy, OPC R&D Manager
- Joe Dunford, Senior Developer
- Jason Fletcher, Project Management
- BJ Dreher, Business Development

1.6 ACKNOWLEDGEMENTS

This project was made possible though the various contributions in cash and in-kind from its participants:

- ***Transport Canada***: the project is 50% funded by Transport Canada, through a Contribution Agreement under Canada's Intelligent Transportation Systems (ITS) R&D Plan *"Innovation Through Partnership"*. The Plan is part of the ITS component of the Strategic Highway Infrastructure Program.

- ***MTQ***: contribution in cash and in kind (equipment and personnel) of up to $75,000.

- ***Tecsult and Matrikon***: contribution in cash and in-kind (personnel).

2.0 WORK DONE AND RESULTS

2.1 WORK PLAN AND METHODOLOGY

The project involved primarily the following major activities:

a) determining the functional requirements of the proposed OPC Server with the MTQ;
b) design and development of the software application by Matrikon;
c) testing of the software application with MTQ provided DMS equipment;
d) test results evaluation and adjustments to the software application;
e) final test site acceptance tests and adjustments;
f) integration in an ITS environment (MTQ ITS pilot project);
g) sign-off by the MTQ;
h) final report to TC.

A Project Execution Plan was produced by Matrikon (DDE-TCS 4502) and is provided as part of Appendix A-1. The project work program identifying the different phases, activities, deliverables and timeline is presented as Appendix A-2 to this document.

2.2 FUNCTIONAL REQUIREMENTS

The functional requirements for the OPC server were determined by a technical committee composed of MTQ specialists actively involved in traffic management and testing of ITS equipment. Tecsult was responsible for managing the relationship and coordinating the activities with the MTQ technical committee.

The document OPC Server for NTCIP - Specific Requirements 1.4 (DDE – TCS 4502) dated 23rd September 2004, produced by Matrikon in cooperation with Tecsult, outlines the specific details of the requirements that had to be met and delivered in the project for the OPC Server for the NTCIP protocol. The document is provided as Appendix B-1.

Transport Canada provided comments on the content of the specifications document. The response to these comments is provided in Matrikon's document OPC Server for NTCIP (DDE-TC4502) as part of Appendix B-2.

2.3 DESIGN AND DEVELOPMENT

The design and development phases were carried out by the Matrikon team during the July to November 2004 period.

The following activities were carried out during that period:

- Preliminary design completed.
- Detailed design completed, reviewed internally and accepted.
- Communication prototype completed and successfully moved data from IDI simulator.

- Development work carried out on the intial prototype development for delivery in November and testing in Montreal in December 2004.

Technical progress achieved during that period:

a) SNMP protocol completed.
b) PMPP and T2 work completed.
c) Agreement on supported NTCIP objects for MTQ completed and designed.
d) Remote configuration interface completed.
e) Software development (beta version) completed. This version included all the MTQ required functionalities except for the PPP function.

2.4 PROTOTYPE DEVELOPMENT

The prototype development was carried out by the Matrikon development team during the September to November time frame as described in section 2.3. A beta version of the OPC server was produced for field testing in Montreal in December 2004.

A NTCIP simulation tester software from IDI was used as a reference during the development phase for testing the OPC server compliance to NTCIP.

The final version of the prototype was developed during the January to March 2005 period. It integrated the missing functionality of the beta version (ie. PPP function) and adjustments that had to be made as a result of the initial testing.

2.5 TEST PLANS AND TESTS

FAT testing was carried out in Matrikon's office in Edmonton throughout the development phase using the referenced IDI NTCIP tester.

The initial field testing was carried out in Montreal in December 2004 with fixed and mobile DMS equipment provided by the MTQ. The beta version of the OPC prototype server was tested on both the Daktronics and Vermac DMS equipment. The communication tests were successful. A test procedure and result report (in French) is provided as part of Appendix C.

After these initial field tests, additional development was carried out to make necessary adjustments to the beta version and to develop and integrate some of the missing functionalities (ie. PPP function) into the server; and, to finalize the prototype.

The final prototype was tested in April using equipment from the MTQ Pont Laviolette project in Trois-Rivières. The tests were successful. Small adjustments were made after the tests.The OPC server was then successfully integrated into the project and is still active and operational to this date.

The official MTQ final acceptance tests were carried out in June in Québec using the Vermac DMS equipment provided by the MTQ. The Daktronics DMS equipment was not available at

the time. The MTQ final acceptance tests were successful and the OPC Server development officially accepted by the MTQ. A test procedure and result report (in French) is provided as part of Appendix D.

Following the MTQ system acceptance, Transport Canada requested that further testing be performed on the Daktronics DMS equipment as set out in the agreed work plan. The TC final acceptance tests were carried out in September and October with the Daktronics DMS equipment at the Daktronics site in Lachenaie, Québec. A test procedure and result report is provided as part of Appendix D.

2.6 RESULTS AND ANALYSIS

2.6.1 Preliminary Site Testing – MTQ Montreal (December 2004)

The results of the preliminary tests on the OPC Server prototype that were carried out in Montreal in December 2004 are provided as part of Appendix D. The overall objective of these site tests was to validate the interoperability of the OPC Server with the controllers of the DMS equipment provided by two manufacturers used by the MTQ. The site tests had to serve also as a basis for the preliminary acceptance by the MTQ of the OPC Server prototype designed and developed by Matrikon.

The results of the initial tests made were conclusive. It was demonstrated that all the NTCIP objects for the DMS, in occurrence the objects for NTCIP 1203 and NTCIP 1201, were supported by the OPC Server for NTCIP. The server, as designed, will allow for the development a unique interface to control the DMS from the different manufacturers.

At the time of the tests, no proprietary object had been defined by the MTQ, thus had not been programmed within the OPC Server. Also, no proprietary objects from the manufacturers of DMS equipment were supported by the OPC server, the objective was to provide uniformed and standardized communication mechanisms with the various DMS regardless of the manufacturer. If certain objects and functionalities must be retained, then it is recommended that the MTQ integrate them as custom objects that the manufacturers will be able to or will have to support.

Regarding the communication profiles, the TCP and IP standards were not validated on site. However, Matrikon confirmed that they tested these functionalities and were validated at their workplace. The PPP standard had not been implemented in the OPC server by the time of these tests. This development work had been planned for the next period.

2.6.2 Additional Development (January-March 2005)

After the initial site tests, there was some additional development work done by Matrikon as planned in the work program in order to fix some of the bugs, do some fine-tuning and adjust some of the features of the OPC Server as per the results of the initial tests and also to enable it to support the PPP function.

Initially, the PPP function was implemented using the default Windows[1] methods and worked. However, during the testing, Matrikon could not get the configuration for Windows to be consistent enough in order to document proper procedures to the MTQ. The complexity of implementing a PPP connection would have then fallen on the client, something that was not acceptable.

Therefore, in the interest of making the product easier to use and more consistent in its implementation, Matrikon decided to change the design for PPP to remove the complexity. This change in approach caused a slippage in the work schedule with the final release of the OPC Server delayed by 3 to 4 weeks.

By early February, all requirements and bugs had been fixed within the OPC Server and Matrikon was able to provide a unit tested version that supported everything except the PPP connections to DMS signs.

2.6.3 Project Testing and Integration – Pont Laviolette (April 2005)

A beta version of the OPC Server, with the PPP functionality, was provided by Matrikon to Tecsult in March for site testing and integration within the MTQ Pont Laviolette Project.

These tasks were successfully carried out in April 2005.

The OPC Server is used to control communication between the Traffic Management Center and the following DMS equipment: Vermac mobile sign.

The OPC Server is working as per the MTQ specifications and so far no performance or technical problems have been reported.

After the completion of the official final acceptance testing adjustments will be made to the software and an official final version will be provided to the MTQ for the Pont Laviolette project.

2.6.4 MTQ Final Acceptance Testing (May 2005)

The final MTQ acceptance tests of the final version of the OPC Server were initially planned to take place in Montreal in late April or early May. However, at the request of the MTQ, the tests were re-scheduled and moved to Quebec City.

The tests were carried out on May 10th, 2005 at the MTQ Centre de signalisation site in Quebec City. For these tests, the MTQ provided us with a Vermac mobile DMS. Their new firmware SIGNCORE V3C had been loaded on their DMS equipment.

[1] Windows is a registered trademark of the Microsoft Corporation (wherever it appears on this document and annexes).

The MTQ could not provide us with a fixed DMS equipment by Daktronics for the tests as originally planned; the equipment was being used by the MTQ elsewhere as part of their Traffic Management System.

A report of the MTQ final acceptance tests (in French) is provided as part of Appendix D.

The tests carried out were successful and conclusive. It was demonstrated that all the NTCIP objects for the DMS equipment provided (ie. objects for the NTCIP 1203 and NTCIP 1201 standards) were supported by the OPC Server for NTCIP.

The OPC server, as designed and developed, is able to provide a unique interface able to control the DMS equipment from various manufacturers. It is not designed however to support proprietary objects from the manufacturer's equipment. This is an issue that has to be addressed between the MTQ and the manufacturers.

With respect to the communications profiles, the OPC Server is designed to support a combination of the following communication standards:

1. SNMP-T2NULL-PMPP-RS232 DIRECT (ie. fibre optics or direct RS232)
2. SNMP-T2NULL-PMPP-RS232 MODEM (ie. standard or cellular)
3. SNMP-TCPIP-PPP-RS232 DIRECT (ie. fibreoptics or direct RS232)
4. SNMP-TCPIP-PPP-RS232 MODEM (ie. standard or cellular)

The profiles 1 and 2 were tested at length on the equipment provided by the MTQ. The profiles 3 and 4 were not tested and could not be validated due to the lack of DMS equipment supporting the standards. These communications profiles supporting TCP and IP are desirable because they will satisfy the networking requirements in a near future and their life expectancy is greater.

Matrikon was able to confirm that the functionalities of profiles 3 and 4 were validated using the reference available test tools in their Edmonton office.

2.6.5 TC Final Acceptance (August 2005)

At the request of Transport Canada, additional testing had to be performed on the fixed DMS equipment provided by Daktronics at the manufacturer's site in Lachenaie, Qc.

These tests were initially scheduled for September but had to be postponed because a minor adjustment to the software had to be done by Matrikon. The tests were finally carried out by Tecsult in October 2005.

The tests on the Daktronics DMS equipment were successful and conclusive. The results are provided and integrated in the test report to the MTQ as part of Appendix D.

2.7 CORRECTIVE MEASURES AND FINAL VERSION

Following the analysis of the results of the final acceptance tests on the Vermac equipment, minor adjustments were made to the OPC server software and a revised final version was tested on the Daktronics equipment. A final version was then officially delivered to the MTQ and is the one that is currently installed in the Pont Laviolette project.

3.0 SUMMARY OF CURRENT TECHNOLOGICAL ISSUES

3.1 PROBLEMS IDENTIFIED OR REMAINING

Proprietary issue with manufacturer's equipment to be dealt with by the MTQ. See section on conclusions and recommendations.

4.0 CONCLUSIONS AND RECOMMENDATIONS

4.1 RESULTS OF WORK DONE VS OBJECTIVES

The overall objective of the project was to provide a common way to communicate to the DMS devices, each typically managed though proprietary software control applications specifically designed by each manufacturer, using a technology already well established within the industrial market called OPC.

The tests performed on the OPC Server were conclusive. It was demonstrated that al the NTCIP objects for the DMS equipment used, in occurrence the NTCIP 1203 and NTCIP 1201, are supported by the OPC Server. The server is designed for the development and use of a unique interface to control the DMS of different manufacturers.

A few minor points were noted during the final testing and adjustments were made by Matrikon to the final version of the software.

With respect to the communication profiles, the OPC Server was designed to support the following combination of standard communications:

1. SNMP-T2NULL-PMPP-RS232 DIRECT (ie. fibre optics, direct RS232)
2. SNMP-T2NULL-PMPP-RS232 MODEM (ie. standard or cellular)
3. SNMP-TCPIP-PPP-RS232 DIRECT (ie. fibre optics, direct RS232)
4. SNMP-TCPIP-PPP-RS232 MODEM (ie. standard or cellular)

The profiles 1 and 2 were abundantly tested on the DMS equipment provided by the MTQ. The profiles 3 and 4 could not be validated because of the lack of DMS equipment supporting the combination of standards. It is important to understand that by all indications, the standards supporting TCP and IP are desirable because they will correspond to future network requirements and their projected life expectancy will be greater. Matrikon confirmed that these functionalities were successfully tested at their site with the available test equipment.

It is important to note that the OPC Server does not support the proprietary objects of the respective manufacturer's equipment. The objective of the project was to attempt to provide a common and standard means of communications with the different DMS regardless of the manufacturer.

If certain objects and functionalities must be retained, then we recommend that the respective provincial and/or state DOT integrate them into their own set of requirements (ministerial or departmental proprietary objects) and require that all manufacturers support them.

The achievement of the objectives was conditional upon some prerequisites. The first prerequisite was the definition of a uniform dictionary and the uniformization of the communication processes for all the DMS equipment. These two objectives are achieved by the NTCIP standard and the OPC server.

The NTCIP standards define well the dictionary of objects but in some cases may leave some room for interpretation. For instance, the object "DMS Message CRC" was interpreted and implemented differently by the two manufacturers of the DMS equipment used in the project. In this case, the bytes that define the object had to be inverted in order for the message to be activated on the respective DMS equipment.

In order to achieve the objective and succeed in achieving a complete uniformity in the management of the DMS equipment through a unique common interface, it is important that the DOT's standardize the use of certain objects and group of NTCIP objects. Because a unique common control interface must manage the DMS equipment in a unified and transparent fashion, we have to expect that a sequence of commands or writing of a variable produce the same effect from one DMS to another.

The tests showed that for the majority of the functionalities, these pre-requisites were observed from one manufacturer to the other.

In order to facilitate the final integration of the DMS from the different manufacturers on a similar communication network, managed by the same control software, it is recommended that certain NTCIP objects be specified to the manufacturer during the manufacturing and supply process.

Primarily, it will be necessary to manage the expectations with respect to the objects of the conformity group "Font Definition". The objects of this group define the **fonts** of the message displayed on the DMS. A control software will expect to find the same fonts defined the same way in the same memory space of the DMS controller. This uniformity will permit the controller to offer a pre-visualization to scale of a message to be displayed. Otherwise it will not be possible to achieve this function.

The recommendation is to ensure, in the supply process of the DMS equipment, that six (6) standard character sets of the MTQ be loaded in the manufacturing process at the position 1 to 6. The policies 7 to 12 must remain modifiable so to provide the possibility of creating new sets of characters. The policies 13 and ensuing may be used by the manufacturers for their own requirements. Some of these parameters that define the policies must also be made uniformed.

The Annex B to the final test report shows a table of the recommendation for the definition of the NTCIP variables for the policies of the characters.

4.2 NEXT STEPS ENVISAGED

4.2.1 Future Development

Matrikon's plan is to attempt to promote this new product for other traffic management systems in North America. Matrikon's solution has been successfully deployed as part of the Colorado Springs Traffic Management Center to provide a unified operator interface.

If the market potential for this solution is confirmed, then Matrikon will consider expanding the scope of the NTCIP standard implemented within the OPC server to include cameras and environmental systems.

Matrikon is also working towards a more active participation within the ITS community.

5.0 SUCCESSES AND BENEFITS

5.1 INTEGRATION OF THE OPC SERVER IN TRAFFIC MANAGEMENT CENTER OPERATIONS

The OPC server was successfully integrated in the following Traffic Management Center Operations:

a) MTQ Pont Laviolette project as described in this report.
b) Colorado Springs Traffic Management Center to provide a unified operator interface to the Skyline VMS equipment over a fiber optic network. This showed that the OPC server that was developed in this project is truly an open server and is able to communicate with other DMS equipment.

5.2 PRESENTATIONS AT CONFERENCES

a) Annual TAC-ATC Conference and Exhibition, Quebec City, September 2004.
b) ITS Canada Annual Conference in Quebec City, May 2005.

APPENDIX A-1
PROJECT EXECUTION PLAN

Tecsult
OPC Server for NTCIP
Project Execution Plan 1.0

DDE-TCS4502

Matrikon
Solutions for Industrial Agility

DDE-TCS4502 – Project Execution Plan

File Name: Project Execution Plan Template.doc

Prepared By:

Matrikon Inc.

Chris Robinson
#1800, 10405 Jasper Avenue
Edmonton, Alberta
T5J 3N4 Canada

Phone: (780) 448-1010 x(4532)
Fax: (780) 448-9191
Email: Chris.Robinson@matrikon.com

Revision History:

Date	Version	Description	Created by	Reviewed by
2004July09	1.0	Initial Draft	Chris Robinson	

Sign-off:

Approved By	Date	Signature

© Copyright 2005, Matrikon Inc.

Document Terminology

Term/Abbreviation	Description
FAT	Factory Acceptance Test
SAT	Site Acceptance Test
LAN	Local Area Network
WAN	Wide Area Network
MTQ	Ministry of transportation Quebec
TC	Transportation Canada
Tecsult	Matrikon's client for the project, one of Canada's most important engineering firms.
Matrikon	Matrikon Inc.
Matrikon OPC	Matrikon's OPC-Compliant servers and clients
MyMatrikon	Matrikon's web-based project monitoring system
OPC	The communication standards published by the OPC Foundation.
OPC Server	A software package that makes data from a system such as RTP available using the OPC-DA standard.
SCADA	Supervisory Control and Data Acquisition, a term used to describe control and monitoring of remote field devices.
DMS	Dynamic Message Sign
NTCIP	National Transportation Communications for ITS protocol.
Definitive Agreement	Contract signed by both the client and Matrikon that specifies the scope of work and the terms and conditions of the project. This may be a purchase order on small projects.
ASC	Actuated traffic Signal Controller
CDPD	Cellular Digital Packet Data
CCTV	Closed Circuit Television
DCM	Data Collection and Monitoring Devices
ESS	Environmental sensor stations
FSK	Frequency Shift Key
IP	Internet Protocol
ITS	Intelligent Transportation Systems
NEMA	National Electrical Manufacturers Association
PMPP	Point-to-Multi-Point Protocol
PPP	Point-to-Point Protocol
SNMP	Simple Network Management Protocol
UDP	User Datagram Protocol

Table of Contents

1 Introduction 1
1.1 Purpose of Document 1
1.2 Project Overview and Goals 1
2 Project Scope 2
2.1 In Scope 2
2.1.1 Client Responsibilities 2
2.1.2 Matrikon Responsibilities 3
2.2 Out of Scope 4
2.3 Project Constraints 4
2.4 Project Assumptions Affecting Scope 4
2.5 Project Deliverables 4
3 Risk Management 6
3.1 Technical Risks 6
3.2 Project Risks 6
4 Project Execution 8
4.1 Project Organization and Stakeholders 8
4.2 Work Breakdown Structure (WBS) 9
4.3 Detailed Work Plan 9
4.3.1 Project Execution Plan 9
4.3.2 Kickoff Meeting 9
4.3.3 Functional Specification 10
4.3.4 Design Execution 10
4.3.5 Development 11
4.3.6 Testing 11
4.3.7 Procurement 12
4.3.8 Equipment Shipping 12
4.3.9 Customer Acceptance 12
4.3.10 On-Site Implementation 13
4.3.11 Site Acceptance 13
4.3.12 Training 13
4.3.13 Documentation 13
4.3.14 Project De-briefing / Knowledge Transfer 13
4.4 Project Timeline 13
4.5 Project Milestones 13
4.6 Project Closeout 14
4.6.1 Project Documentation and Storage 14
4.6.2 Client Acceptance/Turnover 14
4.6.3 Warranty Period 14
4.6.4 Support Plan 15
5 PROJECT CONTROLS 16
5.1 Cost Control 16
5.2 Document Control 16
5.2.1 Document Reference Numbering 16
5.3 Project Communications 17
5.3.1 My Matrikon 17
5.3.2 Progress Reports 18

5.3.3 Progress Meetings 18
5.3.4 Project Correspondence 19
5.4 Issue Resolution Process 19
5.5 Change/Revision Control 19

Table of Figures

Figure 1 – File storage structure **16**
Figure 2 - Overview screen for existing client **17**
Figure 3 - Project-specific Overview screen for existing client **18**
Figure 4 - Decision flow chart for the Scope and Change Management Plan **20**

Table of Tables

Table 1 - Technical Risks **6**
Table 2 - Project Risks **7**
Table 3 - Responsibility Matrix **8**
Table 4 - Project Milestones **14**

1 INTRODUCTION

1.1 Purpose of Document

The purpose of this document is to establish policy and guidelines for Matrikon's project execution, specially in this case for the development of an OPC Server for NTCIP. This document illustrates the parameters of how the project will proceed from inception to closure. It also defines the project scope, deliverables, milestones, stakeholders, and preliminary risks.

This document should be used as a reference to all project stakeholders.

1.2 Project Overview and Goals

Matrikon, Tecsult, MTQ, and TC have joined together to use their expertise, skills, and know how to provide an OPC server interface to a variety of DMS devices using a transportation communication standard called National Transportation Communications for ITS protocol or NTCIP for short. The purpose of this interface will be to enable a standard method of communication to a variety of differing DMS devices from a range of manufacturers. By developing this standard interface, COTS applications for Control and Monitoring can be used to develop a single, common operator interface to be used within any control center.

In short, the goal of this project is to define, design, create, and deploy an OPC server that will provide data access to a range of DMS devices using NTCIP.

2 PROJECT SCOPE

2.1 In Scope

This section defines the project scope. This defines the baseline for the expectations between each of the stakeholders. Project scope is not written in stone, but any time the project deviates from the basic scope outlined in this section, change control procedures can be initiated to better manage those changes. This approach minimizes cost and execution time while maximizing the quality of the deliverables.

The following points are within scope of the project.

- Environment that the OPC Server is to operate within as specified by the functional requirements (ie: Operating system).
- Interoperability with Matrikon's OPC Explorer application and interoperability with the OPC Foundations OPC DA Self Compliancy tool.
- The Interface to the OPC server provided to the OPC Client.
- Behavior of the OPC server in accordance to the OPC DA specification.
- Physical connectivity between the OPC Server node and 2 DMS devices to be chosen by MTQ. This is referring to physical path devices like modems, serial connections, Ethernet connections etc.
- Description of proper operation of the OPC server.
- Any additional libraries that may be used by the OPC Server.
- Training materials for agreed upon training of Tecsult and MTQ staff.

2.1.1 Client Responsibilities

As outlined in the project kick-off process, client responsibilities in this type of project are primarily focused on providing required resources and information. Tecsult, MTQ, and TC are the clients referred to in this section.

The great majority of the work will be delivered in a timely and cost-effective manner only if the client-side resources are available. Each time a time critical task is set out for the responsible stakeholders, a due date must be assigned to that task. Should the time for delivery of the task extend beyond the agreed upon due date, the project may be impacted by as much time that has passed from the beginning of the due date to the time of delivery of that task. The following are the known responsibilities for one or more of the clients of this project.

- Provide project sponsorship – assume a "Stakeholder" position during project execution
- Timely review and approval of required project documentation, specifically meeting minutes, software requirements, and the project design document. Note the review and approval required by each group may vary depending on the document.

- Access to any pertinent equipment or software for the project. Specifically the project will require access to two NTCIP compliant DMS units from two differing manufacturers for a minimum of a 2 week period for the purpose of site acceptance testing.

- Approval for NTCIP simulation package to be used to development and testing.

- Approval for the third party company who will be assigned to perform a compliancy review of the end product on behalf of the stakeholders.

Tecsult specific Responsibilities

This section outlines the responsibilities only for Tecsult.

- Provide key project document leadership such as authoring of specific documents required by the processes defined by Transportation Canada and MTQ.

- Provide representation for the project to MTQ and TC.

- Assistance in planning, definition, and setup for acceptance testing within Quebec. It is always required that the client participate in defining the tests to be conducted for acceptance to ensure the expectations of the client is met upon completion of this task. Also, because of the languages and distances involved in this project a lead role from Tecsult will be required to ensure planning and setup of the SAT goes smoothly.

- Any language translation requirements.

- Technical consultation as needed

2.1.2 Matrikon Responsibilities

- Designation of a Project Manager and a project team.

- Bi-weekly project status reports following Matrikon's standard project reporting format which includes the reporting period start and end dates, period tasks completed, upcoming tasks to be completed, problem or issues outstanding, and any other relevant comments.

- Acquisition of relevant NTCIP specifications.

- Independent verification, validation, and OPC compliance of the product within Matrikon.

- Prompt notification of project risks and scheduling issues whenever they occur.

- Delivery of agreed upon project deliverables.

- Attendance to the on-site tests to be conducted for acceptance of product by stakeholders.

- Providing training of Tecsult staff on the final deliverables.

2.2 Out of Scope

Anything defined in this section is considered to be out of scope for the current budget of this project. Management of out of scope requirements are done through proper change order procedures and keep the project within a pre defined budget of work.

- Any OPC client applications not pre defined as the acceptance test clients are not within scope of this project. This means that new OPC client applications cannot be added to the project as a requirement for interoperability until after the project signoff.
- Any DMS equipment not pre determined as the compliance test subjects are out of scope.
- Any functionality not defined with the requirements of the project are considered out of scope unless a proper change order has been issued. This includes any NTCIP objects or physical paths not chosen during requirements definition and are agreed to be postponed for future development in order to manage project size.
- Commissioning of the final product in any future sites has not be defined as a deliverable for this project and is therefore out of scope.

2.3 Project Constraints

This section outlines anything that can potentially restrict the project development in any way.

- Stationary DMS devices are far too large to be used for development. Therefore a simulation package must be considered for the development portion of the project and integration testing with a sign of this nature will have to wait until acceptance testing.
- The number of NTCIP objects defined is quite large. Because we are building a generic OPC server that may be used by many differing organizations through North America we need to support as many of these objects as possible. However its been noted that the sheer number of these objects could limit the development to a subset. This constraint is set out by budget.

2.4 Project Assumptions Affecting Scope

- We are assuming that the specified simulator will have the same level of compliance to certain NTCIP objects as the pre determined DMS devices.

2.5 Project Deliverables

Deliverables are tangible items which are measurable. They can consist of documents, reports, products, services, and must be clearly defined here. These deliverables will be the focus of the Project execution section.

The following deliverables must be provided to the client at the times defined by the project schedule.

- Meeting minutes from any meetings including any staff external to Matrikon. It is not required for Matrikon to present meeting minutes from any internal only meetings at Matrikon to stakeholders.
- Project Execution Plan and Milestone schedule.

- Functional Requirements Document.
- Factory Acceptance Test Plan (As developed by Matrikon and Tecsult).
- The Bi-Weekly Status reports delivered via email.
- OPC Server package including installer and standard Matrikon users manuals delivered at the end of the project execution on a CD. For now it's been agreed these manuals will be presented in English.
- Training services for Tecsult on proper usage of product.
- Project Completion Notification.

3 RISK MANAGEMENT

As in any project, the project has inherent risks that must be recognized, understood, mitigated by proactive action on the part of the project team. This section summarizes the known risks at the time of the project inception, and outlines the plans for their mitigation.

Project Risks can be categorized in two areas:

- Business/Project
- Technical

3.1 Technical Risks

These are risks associated with the technical solutions to be posed.

The following table identifies the technical risks in the project associated by probability of the risk occurring and the severity of the impact to the project. The management plan for each risk type is identified.

Table 1 - Technical Risks

Risk item	Probability of Occurring	Severity	Management Plan
Object Definitions too large	High	Medium	Reduce required support.
Simulation tools compliance differs from real world DMS devices	Medium	High	Perhaps we can pre-test the DMS signs against the simulation packages once they are both selected and check for compatibility. Failing that, we can modify the definition during SAT. Action will need to be taken if we are forced into option 2.

3.2 Project Risks

These are risks associated with organizational, project implementation, project participant issues. These can be related to resourcing, client responsibilities, supply of equipment, coordination.

The following table identifies the technical risks in the project associated by probability of the risk occurring and the severity of the impact to the project (classed as H=High, M=Medium, L=Low). The management plan for each risk type is identified.

Table 2 - Project Risks

Risk item	Probability of Occurring	Severity	Management Plan
Finding Viable third party compliance	Medium	High	We are required to have the product verified by a third party, but we do not have a company yet. We must start this process immediately to mitigate the risk of not finding one. Backup plan could be to propose that the simulation package used could prove compliance have it run by Tecsult or MTQ.
Cannot find modems defined in requirements	Medium	Medium	There may be more modems then we have access to. Must either restrict the requirements to devices we have or work with MTQ on getting the range we need as soon as possible.
Cannot get correct version of NTCIP specs	Low	High	We can work with Tecsult on ensuring we have the correct versions of the specifications.

4 PROJECT EXECUTION

4.1 Project Organization and Stakeholders

CLIENT CONTACTS

- Name: Eric Bertrand
  - Title: Automation Specialist
  - Direct: (514) 287-8500 x 8335
  - Fax: (514) 287-8643
  - E-mail: Eric.Bertrand@Tecsult.com

- Name: Jean-Pierre Lauzon
  - Title: Consultant
  - Direct: (514) 287-8500 x 2212
  - Fax: (514) 282-2808
  - E-mail: JP.Lauzon@Tecsult.com

MATRIKON CONTACTS

- Name: Chris Robinson
  - Title: Project Manager
  - Direct: (780) 448-1010 x 4532
  - Cell: (780) 232-0744
  - Fax: (780) 448-9191
  - E-mail: Chris.Robinson@Matrikon.com

- Name: Ryan Sadler
  - Title: Project Developer
  - Direct: (780) 448-1010 x 4555
  - Fax: (780) 448-9191
  - E-mail: Ryan.Sadler@Matrikon.com

Responsibility Matrix

Table 3 - Responsibility Matrix

	Chris Robinson	Ryan Sadler	BJ Dreher	Sean Leonard	Eric Bertrand
Account Management			X		
Matrikon Group Manager					
Matrikon Project Manager	X				
Lead Developer		X			
Independent QA				X	
FAT		X			
SAT		X			

4.2 Work Breakdown Structure (WBS)

The work breakdown structure is a very important portion of a project plan.

The work breakdown structure consists of a list of major work which must be executed to achieve the deliverables on a project. They are tangible, and measurable. They can consist of the major headings required for a detailed work plan, and subsequently a detailed project plan.

4.3 Detailed Work Plan

The following section outlines the work plan the project will follow in order to successfully execute the project.

4.3.1 Project Execution Plan

Before any work begins on the project, ensure a careful plan is formulated. That is this document.

This document is required to be reviewed and approved by the stakeholders. For this document Tecsult is the only required stakeholder to approve this document. This is because Tecsult will be issuing another project plan to MTQ and TC using this document as a resource for that information.

4.3.2 Kickoff Meeting

June 26th, 2004

Attendee: Chris Robinson

Duration: 1.5 days

At the time this document was authored the Kickoff meeting had already taken place in Montreal on June 26th as planned.

Objective	*Kick-off* • Identify project team and communication channels • Review and approve communication numbering scheme • Confirm/clarify scope of deliverable • Review and approve project implementation methodology • Identify methodology for the review and approval of control documents. Identify signing authorities. • Review project milestone schedule • Collect all site-specific information necessary for project execution.
Activity	Conduct kick-off meeting. Collect information vital to the project (tag lists, graphic files, business procedures etc.) and conduct interviews with key CLIENT personnel. Construct project plan: qualify scope & objectives specific to the NTCIP project. Create project milestones schedule. Determine project management procedures: identify CLIENT and Matrikon resources. Define communication and sign-off procedures. Define payment vs. milestone schedule.

Quality Checks	Review of project plan & project management procedures: CLIENT and Matrikon sign-off on these elements.
Deliverables	*Kick-off meeting minutes* *CLIENT and Matrikon organizational charts and contact sheets containing key personnel* *Project Execution Plan* and *Milestone Schedule*

4.3.3 Functional Specification

Objective	The objective of the functional specification is to determine what the functional approach is to each individual component of the project and to show how all the parts combined will provide CLIENT with the functionality as identified at the kick-off meeting and during the site survey.
Activity	Define project components: identify what functional components are required and how these components interact. For the NTCIP project, we have the OPC server component. Create architecture and dataflow diagrams. Specify the required hardware components. Analyze functional requirement for each component. Determine what technologies will be used and which need to be developed. Assign Matrikon resources to each component. If necessary, update scope, budget & schedule based on the functional analysis and take corrective actions.
Quality Checks	Internal review of the Specific Requirements Specification by the Matrikon team. Review of the Specific Requirements Specification: CLIENT reviews and accepts (sign-off) the Specific Requirements Specification.
Deliverables	*Specific Requirements document* defines functionality, provides architecture, data flow and business diagrams, determines CLIENT and Matrikon responsibilities, helps to estimate the project schedule.

4.3.4 Design Execution

Objective	The objective of the Design Execution phase is to design in detail how the requirements of each component as identified in the Specific Requirements Specification are going to be satisfied.
Activity	Design Data Acquisition components. Design Data Storage components. Design Data Processing components. Design inter-group components. Design the hardware configuration. Construct Quality Assurance (QA) Test plan for each component: this test document contains all conceivable situations that the component will be exposed to and the expected behavior in that situation. If necessary, update scope, budget & schedule based on the detailed design and take corrective actions.
Quality Checks	Internal review of the Detailed Design documents as prepared by the assigned component resource by the Matrikon team. Review of the Detailed Design documents: CLIENT reviews and accepts (sign-off) the

	Detailed Design documents.
Deliverables	*Status reports*: updates CLIENT on project progress and project milestones. *Detailed Design documents* for each component. *Quality Assurance (QA) Test plan* for each component.

4.3.5 Development

Objective	The objective of the Development phase is to construct the components as identified in the Functional Specification following the design as identified in the Detailed Design phase.
Activity	Construct the NTCIP OPC Server: configure and/or code software following the Detailed Design specifications. Acquire hardware as identified in the Detailed Design specification. Unit Test the OPC Server per functional unit: internally test each functional unit independently. If necessary, revise the QA test plans based on newly discovered conditions of operation for each component. If necessary, update scope, budget & schedule and take corrective actions.
Quality Checks	Internal review of the OPC server for NTCIP unit tests: Matrikon reviews the components individually using the QA test plan for that component. Review of the QA test plans: Client and Matrikon review the (revised) QA test plans.
Deliverables	*Status reports*: updates CLIENT on project progress and project milestones. *QA Test plans*: (revised) testing procedures and plans.

4.3.6 Testing

Any detailed execution steps for plans for development Refer to the overall test plan.

QA Testing

Objective	The objective of the QA Testing is to test the full system as installed on the designated hardware according to the integrated Test plans in Matrikon's staging area. The QA group is independent of the project team and therefore can neutrally accept or deny the products release based on their results.
Activity	Perform QA testing: using the QA test plans for each component, test each component. These tests include load & performance tests. Resolve any deficiency encountered. Update test plans if necessary. Complete the Factory Acceptance Test (FAT) Specification: based on the integrated QA test plans, the FAT test document will be compiled and executed as an "internal FAT". If necessary, update scope, budget & schedule and take corrective actions.
Quality Checks	Internal review of the QA tests: Matrikon reviews the integration tests. Internal review of the FAT specification: Matrikon reviews the FAT specification. Review of the FAT specification: CLIENT and Matrikon review the FAT specification.
Deliverables	*Status reports*: updates CLIENT on project progress and project milestones. *Factory Acceptance Test specification*: document describing the FAT specifications.

FAT Testing

Objective	The objective of the Factory Acceptance Test (FAT) is to determine that the components

	perform as specified in a simulated environment as similar as possible to the site system. The FAT demonstrates the working in Matrikon's staging area and allows CLIENT to provide acceptance.
Activity	Perform factory acceptance test: test components Matrikon's staging area using simulated data. Resolve any deficiency encountered. Update test plans if necessary. Gain acceptance of working Design If necessary, update scope, budget & schedule and take corrective actions.
Quality Checks	Matrikon review and accept the components as built by Matrikon and indicate acceptance to Tecsult. This will ensure that the SAT will operate with reduced risk.
Deliverables	*Status reports*: updates CLIENT on project progress and project milestones. *Factory Acceptance Test sign-off*: indicates in writing that CLIENT has tested the components and accepts these as built by Matrikon.

SAT Testing

Objective	The objective of the Site Acceptance Test (SAT) is to determine that the components perform as specified in the actual working environment. The SAT demonstrates the working Design on site at CLIENT using live data and allows CLIENT to provide acceptance. In this case the SAT will be an onsite execution of the same test plan designed and tested in FAT.
Activity	Perform SAT: test components with CLIENT project team at CLIENT's site using live data. Gain acceptance of working Design: CLIENT verifies and indicates in writing that the Design is functioning as specified. If necessary, update scope, budget & schedule and take corrective actions.
Quality Checks	Signing of the SAT form: CLIENT and Matrikon review and accept the components as installed by Matrikon and indicate acceptance in writing.
Deliverables	*Status reports*: updates CLIENT on project progress and project milestones. *Site Acceptance Test sign-off*: indicates in writing that CLIENT has tested the components at CLIENT's site and accepts these as delivered by Matrikon.

4.3.7 Procurement

At this time no procurement has been identified.

4.3.8 Equipment Shipping

Currently there is no plan to ship any DMS devices to Edmonton, instead a software simulation package will be used. However we may want to consider having modems shipped as they are small and MTQ may have some available for our development. If they are available we will have MTQ manage the shipping details for the equipment.

4.3.9 Customer Acceptance

The stakeholders of the project have decided to use a third party simulation package acceptance test the OPC server. This is due to a lack of an industry standard test application or because there are no known DMS devices that support a complete enough range of NTCIP objects that we can fully test and qualify the full functionality of the OPC server. The proposed test application to be used to simulate the functionality of a DMS device is the

Intelligent Device Simulator provided by MTQ.

This application will be officially accepted as the project test applications by the stakeholders once a review by Matrikon has been made of these tools.

4.3.10 On-Site Implementation

Tecsult and Matrikon will define the on site planning once the project has arrived to the end of the development stage.

4.3.11 Site Acceptance

4.3.12 Training

Matrikon is going to provide Tecsult with training on the OPC server in Montreal during the on site testing. This will include

- Provide standard product training: training on standard Matrikon products.
- Provide operator training: train operations personnel in the use and operation of the deliverables
- Provide engineer training: train engineering personnel in the use and operation of the
- Provide system administrator training: train system personnel in the use and operation of the deliverables.

If necessary we may have to update scope, budget & schedule and take corrective actions should training beyond these items be required.

4.3.13 Documentation

In general is has been agreed that all documentation will be provided in English. The documents to be provided are as follows.

- Standard software component documentation.
- Users Manuals
- Training Manuals

4.3.14 Project De-briefing / Knowledge Transfer

It is standard practice that Matrikon's project teams always do a project closeout review. At that time a lessons learned discussion takes place. The results of this discussion are document in the Project Closeout Review document for future reference. Information that can be reused beyond the project team is submitted to the Knowledge database at Matrikon. This system is accessible by all employee's at Matrikon and provides an easy query mechanism for finding this information.

4.4 Project Timeline

The Project time is outlined in a subsequent Microsoft Project Schedule.

4.5 Project Milestones

Project Milestones are key points in the project cycle that are used to measure progress.

Table 4 - Project Milestones

Milestone:	Timeline
P.O. Received	Begin
Kick-off Meeting Complete	June 29th
Functional Requirements Document	July 23rd
Detailed Design Document	TBD
Coding Cycle	TBD
Quality Assurance & Testing	TBD
User Manual	TBD
FAT Completed and Accepted	TBD

4.6 Project Closeout

As the project nears completion and only minor deficiencies exist, then it is time to proceed with the project closeout procedure. Complete documentation of proposals, requirements, original design, change orders, project history and correspondence is essential. This information is then archived.

At Matrikon a Project Closeout Review meeting is conducted to ensure these operations have been completed and that nothing has been missed. The review meeting covers a set of standard closeout requirements. Once the project closeout meeting has been successfully completed and all outstanding project issues resolved, all stakeholders will be notified via delivery of a project completion notification.

The following sections deal with these requirements

4.6.1 Project Documentation and Storage

All relevant (project information that is to be stored for future use) project documentation is assembled. This documentation includes documentation of original design, change orders, and project history. Any drawings that have been stamped by an engineer are stored (electronically or on paper) with those stamps.

All Project information within the Systems group at Matrikon is stored in two places, one is on a secure network drive that is backed up daily. The second is within MyMatrikon's web documents, another system that provides secure documentation storage and backup.

4.6.2 Client Acceptance/Turnover

Tecsult will accept the product after completion of the SAT testing noting minor deficiencies if required. At this point the project will be considered complete and notification from Matrikon can be submitted to Tecsult for acceptance.

4.6.3 Warranty Period

This project will provide a warranty period of 60 days beginning from the acceptance of the Project Completion Notification. This means that any problems or deficiencies found within the product during this first 60 day period are reported directly to the project manager (Chris Robinson) and are required to be resolved by the end of that period (unless otherwise agreed to another date specified by both Matrikon and Tecsult). Tecsult is entitled to any updates or releases provided during this warranty period. Once the warranty period expires any further issues must be dealt with through the Matrikon Support Plan.

4.6.4 Support Plan

Tecsult will be offered a 1 year support plan beginning from the acceptance of the Project Completion Notification. This entitles Tecsult (or indirectly MTQ) access to

- The Matrikon helpdesk
- Weekly login for proactively monitoring your system
- Remote system audit every year
- Software Updates
- Document Updates
- Email/Bulletin Board Support

Please note that Matrikon does offer support plan extensions on a yearly basis, please contact Matrikon driver's sales for more details.

5 PROJECT CONTROLS

5.1 Cost Control

Cost controls will be managed by the project manager from Matrikon. In the case of this project all expenditures must be approved by the project manager and will fall within the fixed price budget for the project.

5.2 Document Control

For the start of the project, all documents will be deposited in subdirectories with the following tree structure:

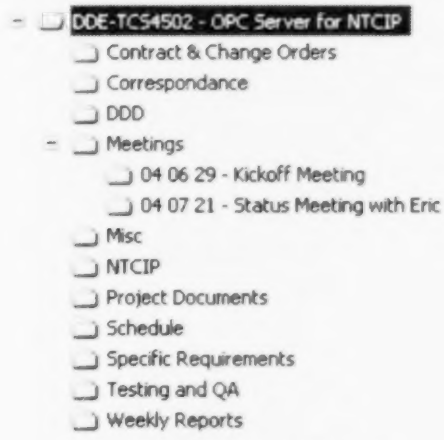

Figure 1 – File storage structure

This structure will be used in MyMatrikon (See 5.3.1) as well as a backup to the files which will reside on MyMatrikon. MyMatrikon will be the central source of officially released documents. The project manager team member from Matrikon will be custodian of MyMatrikon.

Source code for the project will be stored in a revision control system used by Matrikon. This package used is Visual Source Safe. The VSS database is backed up nightly by our automated file server RAID system.

5.2.1 Document Reference Numbering

Typically documents that are shared beyond the project team are numbered using the project code. In this case the file name must include the project code "DDE-TCS4502.X", X denoting the unique ID number for this document in the project. Typically this is incremental from the last document in the document registry.

5.3 Project Communications

5.3.1 My Matrikon

MyMatrikon (http://www.matrikon.com/mymatrikon) is a custom-built product that allows authorized Tecsult personnel to access real-time project information (see Figure). Initially the project stakeholders are provided an account using their email address as their user name and the same for the password. It is recommended that users of MyMatrikon change their password after their first entry into the system.

The information in MyMatrikon will be presented in a clear and concise format through the users desktop, using standard web-browser technology.

MyMatrikon will provide a controlled channel for information exchange between Matrikon and Tecsult. MyMatrikon provides our clients access to internal resources through a world-wide web portal.

Accessible resources include:

- Our support knowledge base,
- Our support call-tracking system,
- Project status and materials,
- Our developer network.

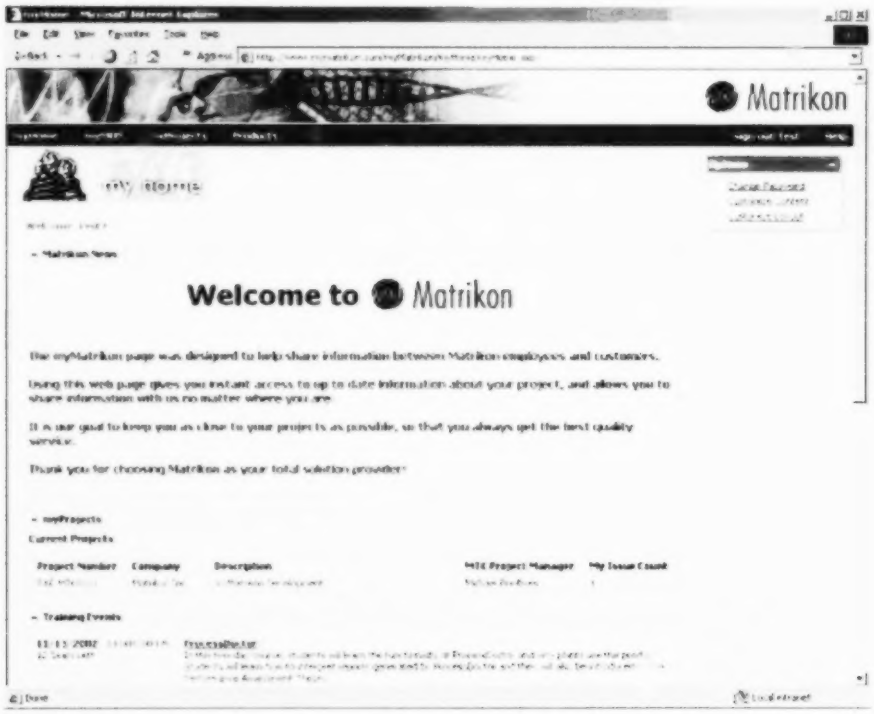

Figure 2 - Overview screen for existing client

During the life of the project, all project materials, correspondence, and development work is available through MyMatrikon. Project managers from both Matrikon and Tecsult can check the status and schedule of the project at anytime, view the latest project updates, submit issues to be resolved, review past issues on the project, and even download and review project documentation or development. The result is a heightened awareness for key project stakeholders of all aspects affecting the success of the project. This promotes collaboration and quick resolution on potential issues that may affect the outcome of the project. Once the project is over, the project materials remain accessible through MyMatrikon throughout the warranty period.

Figure 3 - Project-specific Overview screen for existing client

5.3.2 Progress Reports

A weekly progress report will be submitted to the client and the project team members at the end of each business week. The Project Manager will prepare the weekly progress report and submit. The weekly progress report will give an executive summary of the project status to date and will give details on the major accomplishments of the past week and the tasks planned for the coming week. The report will also provide status of time spent by Matrikon members, expenses to date, total Matrikon project cost to date, project progress.

5.3.3 Progress Meetings

A weekly progress meeting will occur once per week with the core project team members in attendance. If a weekly in-person meeting cannot be executed, then a telephone conference amongst the members will be done. The meeting will serve as a forum to discuss project status and issues. These meetings will be scheduled for Wednesdays.

5.3.4 Project Correspondence

All project correspondence is archived within the Web Documents area of MyMatrikon. It is required that timely acknowledgements be achieved by all stakeholders in regards to any correspondence. This reduces uncertainty and schedule slippage while promoting faster issue resolution. Even if a specific correspondence does cannot be answered immediately, it is required that acknowledgement of the correspondence be returned with an estimation of time on a proper response.

5.4 Issue Resolution Process

The Matrikon project manager will record all issues encountered by all team members in an issue log. This log will reside on My Matrikon. Issues will be recorded with the following categories:

- Project related
- Business related
- Technical related

The issues log will reside on My Matrikon, and viewable by all registered users of the board. On a weekly basis, the Matrikon project manager will review the issues. Issues will be reviewed by Matrikon and the client, during the project status meetings.

Any issues which cannot be resolved by the team and Matrikon project Manager will be escalated to the primary project manager.

5.5 Change/Revision Control

If Tecsult does not agree with the Key Project Document, the document will not be signed off by Tecsult. When both Tecsult and Matrikon agree that Tecsult's desired changes constitute a scope change, a Change Order will be issued to Matrikon by Tecsult and Matrikon will implement the scope change. If Tecsult's desired change is not considered a scope change, the (sub) phase is revisited with the desired changes to produce a revised Key Project Document.

Each Key Project Document-producing (sub) phase is started based on the Specific Requirements as defined in the Definitive Agreement. The Key Project Document that is produced in the (sub) phase is presented for review and sign off by Tecsult. The Signed-Off Key Project Document is considered an amendment of the Definitive Agreement in the event that the Signed-Off Key Project Document amends, extends or contradicts the Definitive Agreement. When the Key Project Document indicates a scope change (a mutual agreement between Tecsult and Matrikon), the requirements are revised and signed off by Tecsult. The revised requirements constitute a Change Order, which becomes part of the Definitive Agreement.

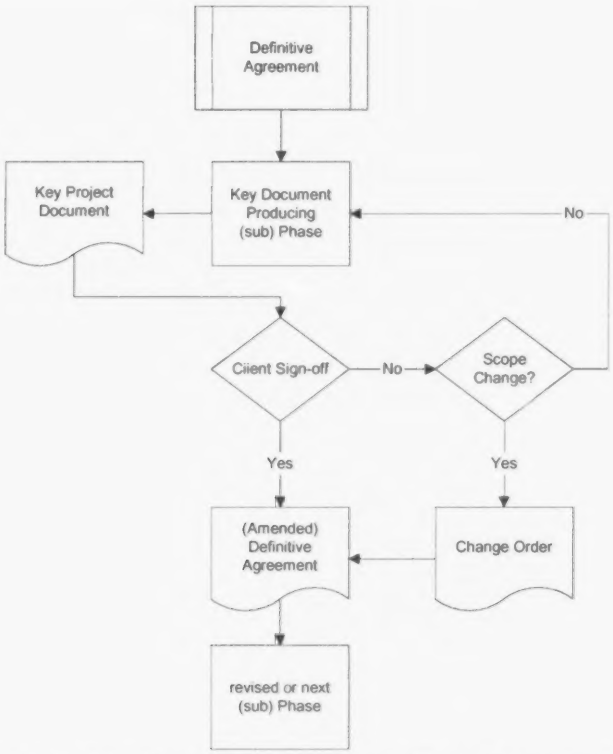

Figure 4 - Decision flow chart for the Scope and Change Management Plan

APPENDIX A-2
PROJECT WORK PROGRAM

PROJECT : OPC SERVER FOR NTCIP FOR ITS EQUIPMENT
SCHEDULE A

TC Reference: 2004RDA-08

PROJECT WORK PLAN	RESPONS.	START DATE	END DATE	Revised End	TIMING	DELIVERABLES
Preliminary administrative work						
Letter of project approval by Transport Canada(TC)/ITS	TC	14.05.04			CA	
Contribution agreement (CA) with TC	Tecsult	14.05.04	30.06.04			Signed agreement.
Annex A - Work Plan	Tecsult	21.05.04	18.06.04			Work plan
Annex B - Budget Cash Flow	Tecsult	21.05.04	18.06.04			Budget
Collaboration agreement Tecsult (T) and Matrikon (M)	T/Mat	21.05.04				Agreement
Licensing agreement Matrikon and MTQ	T/Mat	21.05.04				License agreement
MTQ Contribution (cash)						
Proposals to MTQ	Tecsult	07.06.04	18.06.04		CA + 5w	Proposals to MTQ
Contract award by MTQ to Tecsult (no.1)	T/MTQ	07.06.04	30.07.04		CA + 11w	Contract work and report.
Contract award by MTQ to Tecsult (no.2)	T/MTQ	02.08.04	30.09.04		CA + 7w	Contract work and report.
Billing to MTQ against contracts 1 and 2	T/MTQ	02.08.04	31.03.05		CA + 20w	Contract payment as MTQ contribution
Phase 1 - Planning and Inception						
Review of work package and schedule	T/Mat	21.05.04	11.06.04		CA + 4w	
Preparation of detailed work plan	T/Mat	21.05.04	30.06.04		CA + 7w	Project work plan
Kick-off meeting with MTQ	T/M/MTQ	22.06.04	29.06.04		CA + 6w	Tecsult minutes of meeting.
Initial meeting with TC/TDC	T/M/TC	30.06.04	30.06.04		CA + 7w	TC minutes of meeting.
Project plan + Methodology report to TC	Tecsult	30.06.04	31.07.04		CA + 11w	**TC deliverable - report no.1**
Phase 2 - Functional Requirements						
Draft preliminary functional requirements	Tecsult	21.05.04	18.06.04		CA + 5w	Preliminary functional requirements document (PFRD)
Review of functional requirements	T/M	21.06.04	30.06.04		CA + 6w	Review and comments by Matrikon. Revised functional requirements doct.
Convene MTQ Technical Committee	T/M/MTQ	22.06.04	29.06.04		CA + 6w	Minutes of discussion of functional requirements (PFRD)
Draft functional requirement document	M/T	29.06.04	23.07.04		CA + 10w	Draft funcitonal specs. document
Final review of functional requirements doct. by MTQ	T/MTQ	23.07.04	29.07.04		CA + 11w	Final functional requirements doct. (FFRD)
Approval of functional requirements doct.	MTQ/T	29.07.04	29.07.04		CA + 7w	Approved final functional requirements document
Report - Functional Specifications document	T/M	29.06.04	06.08.04		CA + 12 w	**TC deliverable - report no.2**
Phase 3 - Design and Development						
Preliminary design	Matrikon	02.08.04	13.08.04		CA + 13w	Preliminary design document based on PFRD
Detailed design (with approved FSD)	Matrikon	16.07.04	31.08.04		CA + 16w	Final design document based on FFRD
Start development work	Matrikon	16.08.04			CA + 10w	
OPC Server development work	Matrikon	16.08.04	15.10.04		CA + 22w	
Preliminary version (prototype)	Matrikon		15.10.04		CA + 22w	Prototype
Status report - mid project	T/M		10.02.05		mid-term	**TC deliverable - report no.3**
Meeting - TC Project Review Committee	Tecsult		10.02.05		mid-term	part of TC deliverable - phase 3. Invoice no.3 to TC.
Phase 4 - Testing						
Coordination with MTQ for field test	T/M/MTQ	05.07.04	30.07.04		CA + 11w	
Training - OPC Server	T/M/MTQ					
- Planning of session	T/M/MTQ	11.10.04	15.10.04			
- User training	T/M/MTQ	18.10.04	22.10.04		CA + 23w	
Preparation of a basic application	T/M/MTQ	06.09.04	15.10.04			
Preparation of a test plan	T/M/MTQ	06.09.04	24.09.04			
Factory Acceptance Testing (FAT)	Matrikon	27.09.04	30.11.04		CA + 28w	MTQ PC-based DMS simulator + mobile DMS
Compliance Testing NTCIP (independent firm)	Matrikon	27.09.04	tbd			NTCIP compliance test report
Site Acceptance Testing (SAT) in Montreal	T/M/MTQ	11.10.04	17.12.04		CA + 31w	MTQ PC-based DMS simulator + mobile DMS. Test report.
Code adjustments, re-tests and and completion	Matrikon	04.01.05	31.01.05		CA + 37w	MTQ test environment. Test report.
Completion of PPP tests with DMS sign	T/M/MTQ			08.04.05		
Software final version for MTQ acceptance tests	Matrikon			18.04.05		
MTQ final acceptance tests	T/M/MTQ	21.02.05	11.03.05	06.05.05	CA + 43w	
Matrikon QA internal acceptance process	Matrikon	11.04.05	18.04.05			
Software packaging and final delivery to MTQ	Matrikon	25.02.05	18.03.05	13.05.05	CA + 52w	Software and documentation.
Product documentation and support instructions	Matrikon	25.02.04	18.03.05	22.04.05	CA + 49w	Product documentation and user instructions
Status report - end of project	Tecsult	07.03.05	18.03.05	13.05.05	CA + 52w	**TC deliverable - report no.4**
Meeting - TC Project Review Committee	Tecsult		tbd			part of TC deliverable - phase 4
Draft final report	Tecsult	07.03.05	31.03.05		CA + 45w	part of TC deliverable - phase 4
Final report and project summary	T/M	07.03.05	15.04.05	27.05.05	CA + 54w	**Final report and project summary. TC deliverable report no.5.**

TECSULT

APPENDIX B-1
SPECIFIC REQUIREMENTS

Tecsult / MTQ
OPC Server for NTCIP
Specific Requirements 1.4
DDE-TCS4502

Matrikon
Solutions for Industrial Agility

DDE-TCS4502 – Specific Requirements Document

File Name: OPC Server for NTCIP SRD (DDE-TCS4502.2).doc

Prepared By:

Matrikon Inc.

Ryan Sadler
#1800, 10405 Jasper Avenue
Edmonton, Alberta
T5J 3N4 Canada

Phone: (780) 448-1010 x(4555)
Fax: (780) 448-9191
Email: Ryan.Sadler@Matrikon.com

Revision History:

Date	Version	Description	Created by	Reviewed by
04-07-15	1.0	Document Authoring	Ryan Sadler	Chris Robinson
04-07-27	1.1	Changes based on review with Eric Bertrand from Tecsult.	Ryan Sadler	Chris Robinson
04-08-11	1.2	Version Syncronization	Chris Robinson	Ryan Sadler
04-09-15	1.3	Updates based on feedback from MTQ	Ryan Sadler	Chris Robinson
04-09-22	1.4	Continued updates based on discussions internally spawned from feedback by Tecsult and MTQ.	Chris Robinson	
04-10-13	1.5	Added licensing information for NetSNMP library. Added a diagram describing the transport profiles. Removed inconsistent remarks concerning multiple clients and write serialization.	Ryan Sadler	

Sign-off:

Approved By	Date	Signature

© Copyright 2005, Matrikon Inc.

Document Terminology:

Term/Abbreviation	Description
FAT	Factory Acceptance Test
SAT	Site Acceptance Test
LAN	Local Area Network
WAN	Wide Area Network
MTQ	Ministry of Transportation of Quebec
TC	Transportation Canada
Tecsult	Matrikon's client for the project.
Matrikon	Matrikon Inc.
Matrikon OPC	Matrikon's OPC-Compliant servers and clients
MyMatrikon	Matrikon's web-based project monitoring system
OPC	The communication standards published by the OPC Foundation.
OPC Server	A software package that makes data from a system such as RTP available using the OPC-DA standard.
SCADA	Supervisory Control and Data Acquisition, a term used to describe control and monitoring of remote field devices.
DMS	Dynamic Message Sign
NTCIP	National Transportation Communications for ITS protocol.
Definitive Agreement	Contract signed by both the client and Matrikon that specifies the scope of work and the terms and conditions of the project. This may be a purchase order on small projects.
ASC	Actuated traffic Signal Controller
CDPD	Cellular Digital Packet Data
CCTV	Closed Circuit Television
DCM	Data Collection and Monitoring Devices
ESS	Environmental sensor stations
FSK	Frequency Shift Key
IP	Internet Protocol
ITS	Intelligent Transportation Systems
NEMA	National Electrical Manufacturers Association
PMPP	Point-to-Multi-Point Protocol
PPP	Point-to-Point Protocol
SNMP	Simple Network Management Protocol
UDP	User Datagram Protocol
MIB	Management Information Base

References:

NTCIP Reference	Doc number	Title	Status and amendment
1101	NEMA TS 3.2-1996	*STMF : Simple Transportation Management Framework*	Approved Amendment 1
1102	OER	Octet encoding Rules	V01.11 (Recommended)
1103	STMP	Transportation Management Protocol	
1201	NEMA TS 3.4-1996	*Global Object Definitions*	V02.24 (Recommended)
1203	NEMA TS 3.6-1997	*Object Definitions for Dynamic Message Signs*	V01.15 (Approved) Amendment 1 (Approved)
2101	SP-PMPP/RS232	*Subnetwork Profile for Point to Multi-Point Protocol usng RS-232*	V01.19 (Approved)
2102	SP-PMPP/FSK	*Subnetwork Profile for Point to Multi-Point Protocol using FSK modems*	V01.08 (Recommended)
2103	SP-PPP/RS232	*Subnetwork Profile for Point-to-Point Protocol using RS 232*	V01.13 (Recommended)
2201	TP-Null	Null Transport Profile	V01.13 (Recommended)
2202	TP-Internet	*Internet Transport Profile*	V01.05 (Approved)
2301	AP-STMF	*Simple transportation Management Framework*	V01.08 (Approved)
Data Access Custom Interface Standard			Version 2.05
Data Access Custom Interface Standard			Version 3.00

Table of Contents

1 Introduction 1
1.1 Project Overview 1
1.1.1 NTCIP 1
1.2 SCADA Framework 2
1.2.1 Deployment 3
1.2.2 Data Flow 3
1.3 System Environment 6
1.3.1 External Software System Overview 6
1.3.2 External Hardware Overview 7
2 Specific Requirements 8
2.1 Functional Requirements 8
2.1.1 General NTCIP Requirements 8
2.1.2 General OPC Requirements 8
2.1.3 Installation Requirements 10
2.1.4 OPC Group Requirements 10
2.1.5 OPC Item Requirements 10
2.1.6 Browsing and Item Properties 10
2.1.7 OPC DA 2.0 I/O Operations 11
2.1.8 OPC DA 1.0 I/O Operations 12
2.1.9 Object Hierarchy 12
2.1.10 Common object communication requirements 13
2.1.11 Communication object Statistics 13
2.1.12 Network Channel Requirements 13
2.1.13 Host Connection Requirements 14
2.1.14 Serial Channel Requirements 14
2.1.15 Radio Link object Requirements 14
2.1.16 Dial-up object Requirements 14
2.1.17 DMS Device object Requirements 15
2.1.18 Configuration and User Interface 15
2.1.19 Logging 16
2.1.20 NTCIP DMS Object Requirements 16
2.1.21 DMS Property Management requirements 16
2.2 Non-Functional Requirements 17
2.2.1 Software Interface Requirements 17
2.2.2 OPC Interfaces 18
2.2.3 Common Interfaces 18
2.2.4 Communication Interface Requirements 18
2.2.5 Documentation Requirements 19
3 Requirements Table and Traceability 20

List of Appendices

NetSNMP Copying Agreement 21

Table of Figures

Figure 1-1. Deployment 3
Figure 1-2. API Message Flow 4
Figure 1-3. General OPC Message Flow 5
Figure 1-4. General OPC Architecture 5
Figure 1-5. Scanning the device and updating the client 6
Figure 1-6. Reads from Cache 6
Figure 1-7. Reads from Device 6
Figure 2-1. Sample Object Hierarchy 12

Table of Tables

Table 1. Requirements Cross Reference Matrix 20

1 INTRODUCTION

This document outlines the specific details for the requirements that must be met and delivered in the project for the OPC Server for the NTCIP protocol. This document is reviewed by all the stakeholders and signed off as an official agreement for this functionality in the final deliverables. Only the functionality defined using the "Must" clause is required to be upheld in the final deliverables. This first chapter provides some background to the reader on the project but does not contain any accountable specific requirements.

1.1 Project Overview

The OPC server for NTCIP is an OPC server that will be designed to gather data from a variety of various NTCIP compliant DMS devices. These devices (Dynamic Message Signs) are used throughout metropolitan areas in order to provide traffic information to passing motorists. These signs can be both stationary and mobile and are accessed through many different communication media including dial-up modems, fiber optic Ethernet, cellular technologies etc. This traffic information is critical in assisting in various traffic control operations as it can redirect traffic to use alternate routes or warn motorists of potential upcoming hazards.

Communication to DMS devices is typically managed through proprietary software control applications specifically designed by each manufacturer. In a multi-device system this can be a real problem as control room operators must have knowledge as to which sign must be configured, who the manufacturer is of that sign, and which control package is used to deal with that sign. Finally, the interface and operation of these control packages are quite diverse so operators must be trained on each one which requires a lot of extra time and money.

The goal of this project is to provide a common way to communicate to these DMS devices using a technology already well established within the industrial market called OPC. The intention is once the data is represented in this generic way, any OPC enabled control system package can then be used to setup, configure, and monitor any type of NTCIP compliant DMS device. Now operators can all use the same software control package to manage any type of sign which will reduce overall training requirements, confusion, and enhance the abilities of the operators by enabling them to customize the interfaces in ways that make these management tasks easier to deal with.

Because NTCIP is a standard that is used by more than just DMS vendors, it is a hope that in the future this OPC server technology can be expanded to include much more and provide the same benefits and solutions to other segments within the transportation industry.

1.1.1 NTCIP

Figure 1-1 gives an overview of the scope of the NTCIP protocol. The scope of this project includes a subset of the overall NTCIP series of standards.

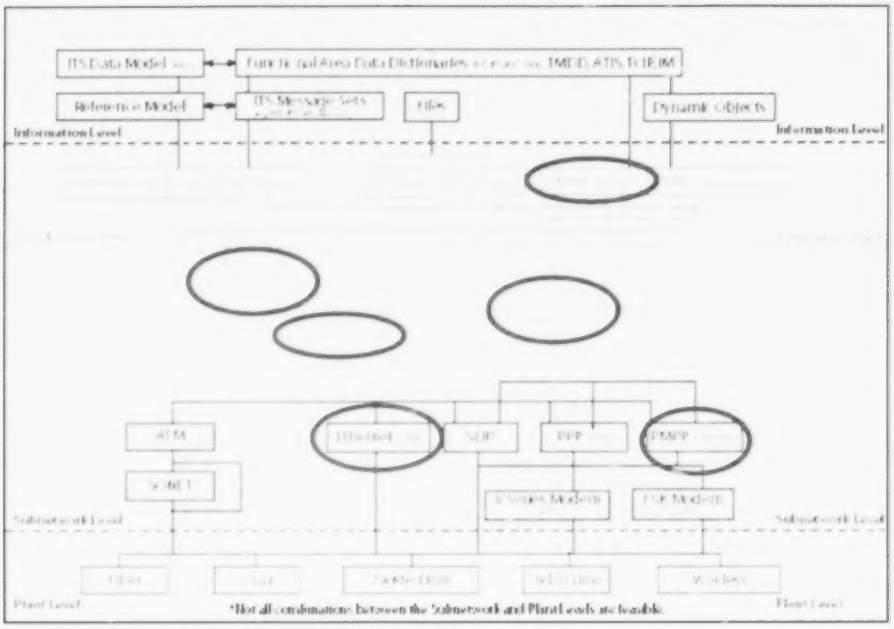

Figure 1-1: NTCIP Standards Framework

1.2 SCADA Framework

Matrikon has two frameworks they can use during the development of any new OPC server interface. One is designed for plants, the other for SCADA systems. Plant edition OPC servers are the more common and are designed to have good, dependable communication with lots of bandwidth, the typical case for a LAN environment. The SCADA framework is designed for SCADA systems that can have poor communication over unreliable connections like radio, dial-up, etc. The SCADA framework is really an enhanced plant framework as it supports everything the plant framework can do and more. Externally the OPC servers built with either of these frameworks appear identical except for the additional configuration capabilities of the SCADA server (adding modems, radios, etc). Due to the requirements of this project the SCADA framework would be a better fit. Finally, there are no differences in licensing for this product.

1.2.1 Deployment

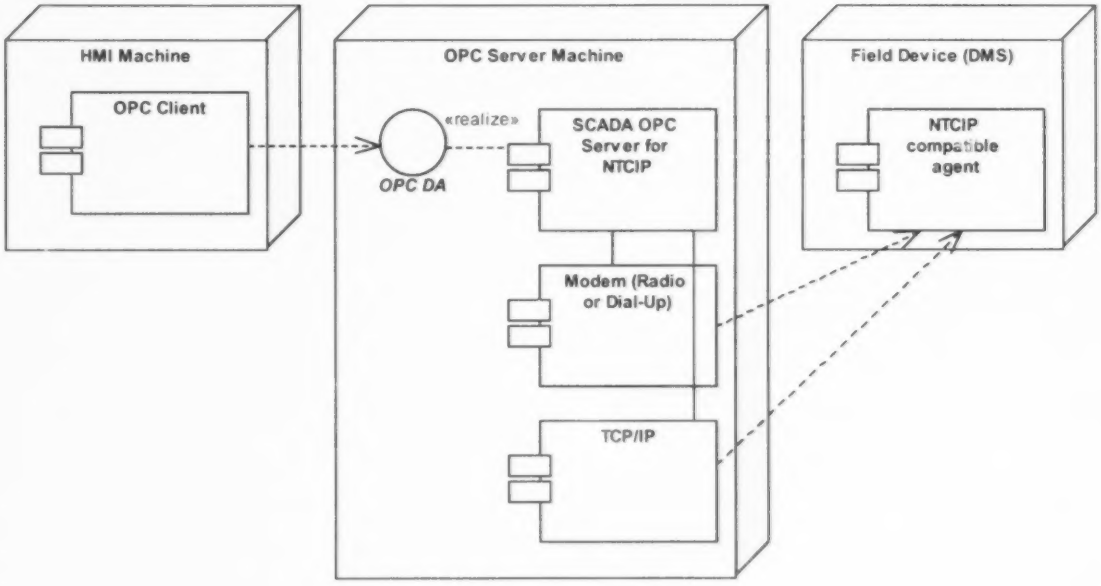

Figure 1-2. Deployment

In the final deployment, an HMI will use the SCADA OPC Server for NTCIP to connect to NTCIP compatible field devices. Communication may occur via TCP/IP over Ethernet or Modem link.

1.2.2 Data Flow

The Server communicates with the device by using Matrikon's SNMP API (Application Programming Interface). This API is responsible for creating and interpreting the messages sent-to and received-by the communications channel.

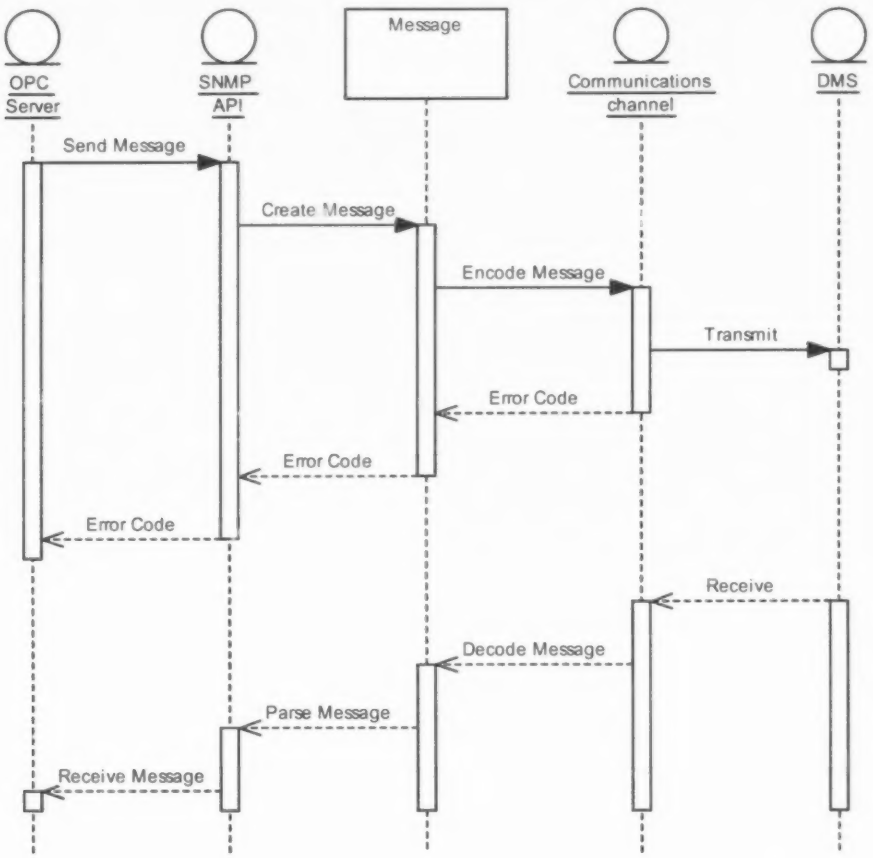

Figure 1-3. API Message Flow

As messages are received, the OPC Server will update its internal cache of data values and update its list of statistics. Data writes to the OPC Server (from the OPC Client) will be sent to the device as demand operations. The OPC server will allow multiple clients to access the various parameters within an NTCIP object, but during write executions it will ensure that there are no interleaving issues with multiple write requests.

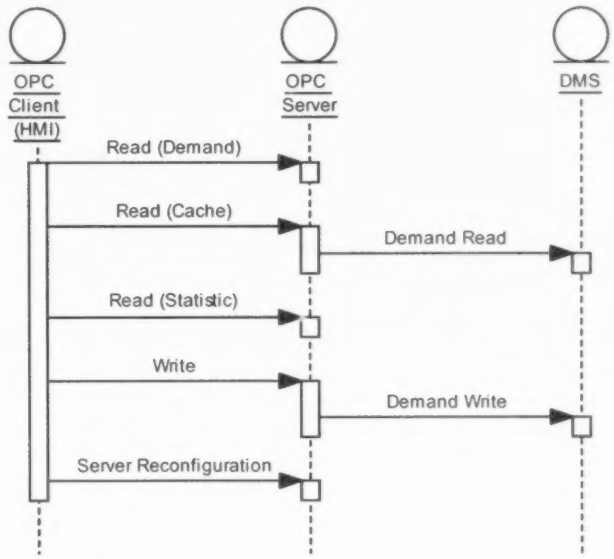

Figure 1-4. General OPC Message Flow

1.2.2.1 OPC Overview

It is useful at this point to discuss some basic terminology relating to how an OPC client can read and write data, and how the OPC server reads and writes data in response to an OPC client's requests. A general architecture for an OPC Client, OPC server, and a Device appears as follows:

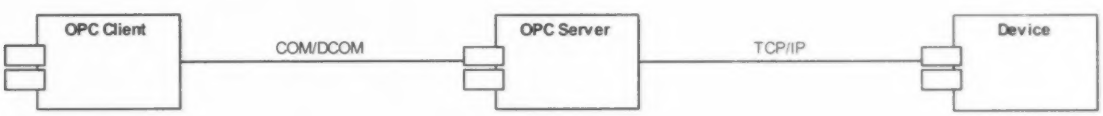

Figure 1-5. General OPC Architecture

There are three ways that an OPC client can read data in the above system: **subscription, reads from cache, and reads from device**. There is only one way a client can write: always to **device**.

1.2.2.2 Subscription

An OPC client can "subscribe" to information in the OPC server. Usually, the OPC client subscribes to a number of OPC items in the OPC server, where each OPC item usually represents a particular piece of data available in the device (such as a process variable). Once the client has subscribed, the OPC server will do two things:

1. The OPC server continually polls the device for new data represented by the OPC items at the update rate requested by the client. The server records the values received in it's memory, or **cache**. This process is called **scanning**.

2. The OPC server informs the client of any changes in the quality or value of the items at a rate no faster than the requested update rate. This is called **updating**. If no changes take place, no updates take place.

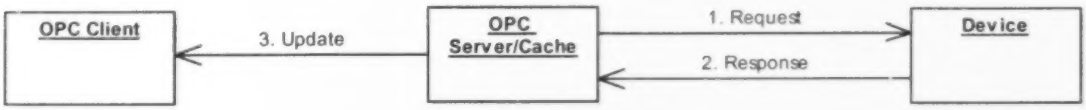

Figure 1-6. Scanning the device and updating the client

Scanning and updating continues until the client **unsubscribes** from the OPC items.

1.2.2.3 Cache Reads

An OPC Client can perform reads from the OPC server's cache. That is, the OPC Client can retrieve a **copy** of the last value the server received from the device. This does not cause the server to make any requests of the device.

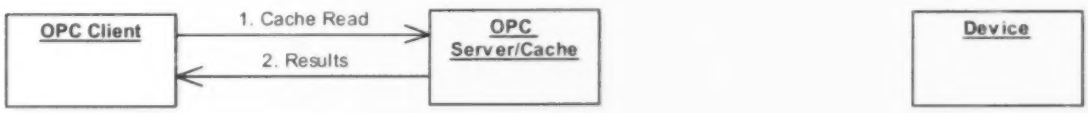

Figure 1-7. Reads from Cache

1.2.2.4 Device Reads

An OPC Client can perform a read operation on demand, meaning that it can force the OPC server to request a value from the device, update its cache, and return the value to the client. Unlike scanning, this is a one-time operation and the value is returned whether or not it is different from the previous value.

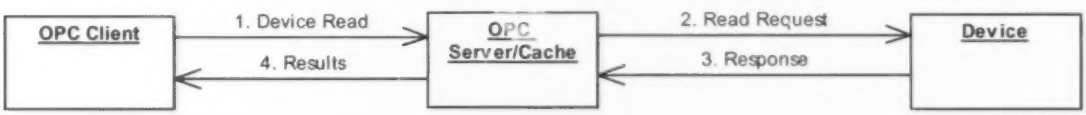

Figure 1-8. Reads from Device

1.3 System Environment

The typical environment the OPC Server will be running in will be on a server at Traffic Control headquarters communicating to a large number of DMS devices through a number of different channels of communication. The OPC client application is not know at this time.

1.3.1 External Software System Overview

1.3.1.1 Operating System

The computer on which the OPC server resides must support the following *minimum* software requirements:

- Microsoft Windows 2000 with Service Pack 2, or
- Microsoft Windows XP with Service Pack 1
- Microsoft XML parser version 4.0
- Microsoft .NET framework version 1.1

There are no other known software applications required to provide normal operation of the OPC server. The XML parser and the .NET framework is distributed with the OPC server and is an option to install during installation.

1.3.1.2 Licensing

The server uses a portion of the NetSNMP library. All rights to the NetSNMP software have been granted without fee provided that a copyright notice is added to any documentation associated with the product.

See Appendix A for the full text of the licensing information associated with the NetSNMP Library.

1.3.1.3 API/Libraries

As noted above, all the communication to NTCIP will be written from scratch or will use source code already available to Matrikon. Therefore there are no external, third party libraries/API's required for this project.

1.3.1.4 Test Applications

The stakeholders of the project have decided to use a third party simulation package to develop and test the OPC server. This is due to a lack of an industry standard test application or because there are no known DMS devices that support a complete enough range of NTCIP objects that we can fully test and qualify the full functionality of the OPC server. The required test application to be used to simulate the functionality of a DMS device is the "IDI Tester"

We may also use the Protocol Analysis tool from FTS, which can determine if the messaging for the OPC server is within specifications.

These applications will be officially accepted as the project test applications by the stakeholders once a review by Matrikon has been made of these tools.

1.3.2 External Hardware Overview

The computer on which the OPC server resides must support the following *minimum* hardware requirements:

- Intel Pentium 4 1.2Ghz
- 256MB RAM
- 1 Ethernet Card
- 1 available serial port
- Dial-up Modem (optional)
- Radio Modem (optional)

Please note that some of the hardware requirements are dependant on how the OPC server is expected to be configured.

2 SPECIFIC REQUIREMENTS

A requirement can be any need or expectation for a system or for its software. Requirements reflect the stated or implied needs of the customer, and may be market-based, contractual, or statutory, as well as an organization's internal requirements. This section contains the actual requirements for the OPC server for NTCIP. Anything stated in this section with a "must" statement will be present in the final deliverables for the project.

2.1 Functional Requirements

2.1.1 General NTCIP Requirements

2.1.1.1 The server must support all Global Objects

This is as defined in standard MIBs and the recommended NTCIP 1201v02.26 standard.

Standard MIBs are available through the NTCIP.org website. The Standard MIBs for the Global Object Definition conform to the NTCIP 1201 v02.26 approved standard. In general, if an object does not exist on the remote side, then and error will be returned. However on the OPC server side, all MIB object definitions are supported.

2.1.1.2 The server must support all DMS Object Definitions

This is as defined in standard MIBs and the approved NTCIP 1203v01.15 standard.

2.1.1.3 The server must support the NTCIP Application Level protocol SNMP

As defined in the approved NTCIP 2301 v01.08 specification.

2.1.1.4 The server must support the NTCIP Transport Level protocol T2/NULL

As defined in the recommended NTCIP 2201 v01.14 specification

2.1.1.5 The server must support the NTCIP Transport Level protocol TCP/IP

As defined in the approved NTCIP 2202 v01.05 specification

2.1.1.6 The server must support the NTCIP Subnetwork Level specifications for PPP over RS-232

As defined in the recommended NTCIP 2103 v01.13 specification.

2.1.1.7 The server must support the NTCIP Subnetwork Level specifications for PMPP over RS-232

As defined in the approved NTCIP 2101 v01.19 specification.

2.1.1.8 The server must support the NTCIP Subnetwork Level specifications for PMPP using FSK Modems

As defined in the recommended NTCIP 2102 v01.08 specification.

2.1.2 General OPC Requirements

This section discusses the general requirements that are not specific to any particular part of the functionality of the OPC server but more related to it overall functionality. In this section any reference to "server" is assumed to mean the "OPC server".

2.1.2.1 The server must run as an out-of-process COM server

In order to support DCOM, the OPC server must be able to be run as an out-of-process COM server. This enables OPC Client applications to access this server remotely.

2.1.2.2 The server must run optionally as an NT/2000 service or as a regular process

In order to maximize the flexibility of the OPC server, it should be designed to run as both a Windows NT/2000 service or as a regular Windows process. This also allows the server to be more tightly tied to a PC and prevents the user from needing to restart the server should the PC require a power cycle or other restart operation.

2.1.2.3 The server must use /SERVICE as the command line parameter to toggle registration

The OPC server must use the "/SERVICE" command line parameter to toggle the registration of the server to Windows. So, if the server is not currently a windows service, running the OPC server executable with the "/SERVICE" command line parameter will make it a service. Conversely, if the server is already running as a service, running the OPC server with the same command line parameter of "/SERVICE" will deregister the server.

2.1.2.4 The server must provide a command line option for registration and deregistration

OPC servers are based on COM components which require registration to the operating system. This function must be provided as a command line option so that the server can be registered or un-registered by installation scripts or other applications without running an instance of the OPC server itself.

2.1.2.5 The server /REGSERVER as the command-line option for registration

As stated in requirement 2.1.2.5, the server must accept "/REGSERVER" as the command line parameter to initiate registration.

2.1.2.6 The server /UNREGSERVER as the command-line option for deregistering

As stated in requirement 2.1.2.5, the server must accept "/UNREGSERVER" as the command line parameter to remove any registration information from the Windows registry.

2.1.2.7 Server must register the OPC sub-key and vendor information string

2.1.2.8 Server must be assigned a globally unique application ID number

2.1.2.9 Server must be assigned a globally unique class ID number

2.1.2.10 Server must register the full path of 32-bit local server executable

2.1.2.11 The server must register any implemented standard component categories

OPC Servers must register their abilities for supporting various OPC specifications through standard component categories. The OPC foundation has defined these categories and therefore the OPC server must register any categories it wishes to provide support for. An example of this would be OPC Data Access V 3.0.

2.1.2.12 Version-independent program ID is Matrikon.OPC.NTCIP

Every OPC server must provide a unique program ID.

2.1.2.13 Version-specific program ID Matrikon.OPC.NTCIP.1

Every OPC server must also register a version independent program ID.

2.1.3 Installation Requirements

2.1.3.1 The OPC server must come packaged in a self installing executable

2.1.3.2 The installer must register all COM based components

2.1.3.3 The installer must be able to uninstall all components and un-register any registration

The OPC server comes with an executable that will perform all the un-registration required.

2.1.4 OPC Group Requirements

2.1.4.1 The server must support the addition, removal, and cloning of *private* groups.

2.1.4.2 The server must allow the client to manage group state for Name

2.1.4.3 The server must allow the client to manage group state for Update Rate

2.1.4.4 The Update rate must be in millisecond resolution

2.1.4.5 The server must allow the client to manage group state for Active State

2.1.4.6 The server must allow the client to manage group state for Time Bias

2.1.5 OPC Item Requirements

2.1.5.1 The server must support the addition, removal, and validation of items in a group.

2.1.5.2 The server must allow the client to manage item configuration for Active state

2.1.5.3 The server must allow item configuration for the requested data type

2.1.5.4 The server must support the following SNMP types: Integer, Sequence, Object Identifier, Octet String

2.1.5.5 An item will successfully validate, if it is found in the specified MIB for the device

2.1.5.6 Items associated with read-only Object Definitions will not permit OPC writes

2.1.5.7 Access rights are determined by the Object Definition in the associated MIB

2.1.5.8 The server will timestamp OPC items with the time at which the cache is updated

NTCIP does not support timestamps on individual requests.

2.1.6 Browsing and Item Properties

Browsable items available will include the Configuration and Control Items associated with that node as described in this document, and items corresponding to items provided in a device MIB.

2.1.6.1 The server must support hierarchical address space browsing

2.1.6.2 The server must support flat address space browsing

2.1.6.3 The server must support branch/leaf name filtering

2.1.6.4 The server must support item data type filtering

2.1.6.5 The server must support item access rights filtering

2.1.6.6 The server must support Value, Quality, and Timestamp properties

2.1.6.7 The server must support the Canonical data type

2.1.6.8 The server must provide the access rights property

2.1.6.9 The server must provide the scan rate property

2.1.7 OPC DA 2.0 I/O Operations

The following requirements are related to the OPC DA 2.0x specification.

2.1.7.1 The server must support OPC 2.0x synchronous reading item values from cache

2.1.7.2 The server must support OPC 2.0x synchronous reading item values from the device

2.1.7.3 The server must support OPC 2.0x synchronous writing control values to items

2.1.7.4 The server must support OPC 2.0x asynchronous reading item values from the device

2.1.7.5 The server must support OPC 2.0x asynchronous writing control values to items

2.1.7.6 The server must support an OPC 2.0x refresh of active items from the cache

2.1.7.7 The server must be able to cancel outstanding OPC 2.0x I/O operations

2.1.7.8 The server must be able to enable or disable OPC 2.0x event notifications

2.1.7.9 The server must support the OPC 2.0x data change event notification client-side interface

2.1.7.10 The server must send an OPC 2.0x notification to the client when a data change is detected

2.1.7.11 The server must send an OPC 2.0x notification to the client when the dead-band is exceeded

2.1.7.12 The server must send an OPC 2.0x notification when an Async refresh operation completes

2.1.7.13 The server must send an OPC 2.0x notification when an Async read operation completes

2.1.7.14 The server must send an OPC 2.0x notification when an Async write operation completes

2.1.7.15 The server must send an OPC 2.0x notification when an Async cancel operation completes

2.1.8 OPC DA 1.0 I/O Operations

2.1.8.1 The server must support OPC DA 1.0a asynchronous reading item values from cache

2.1.8.2 The server must support OPC DA 1.0a asynchronous reading item values from device

2.1.8.3 The server must support OPC DA 1.0a asynchronous writing control values to items

2.1.8.4 The server must support an OPC 1.0a refresh of active items from the cache Read items values

2.1.8.5 The server must be able to cancel outstanding OPC 1.0a I/O operations

2.1.8.6 The server must support the OPC DA 1.0a data change event notification client-side interface

2.1.8.7 The server must send an OPC DA 1.0a notification when a data change is detected

- Data change or dead-band exceeded – data *with* time stamps
- Data change or dead-band exceeded – data *without* time stamps

2.1.8.8 The server must send an OPC 1.0a notification when an Async cancel operation completes

2.1.9 Object Hierarchy

The object hierarchy of the OPC server usually represents a "path" to follow to get to the device. For example, communicating to a hardware device with an Ethernet interface might require a "TCPIP Port object" and a "device" object to be created, configured, and enabled/disabled by the user. Because these objects are specific to a particular OPC server, they must be described and implemented by the specific inner layer. The following figure shows an example of what an Object hierarchy may look like for the NTCIP OPC Server.

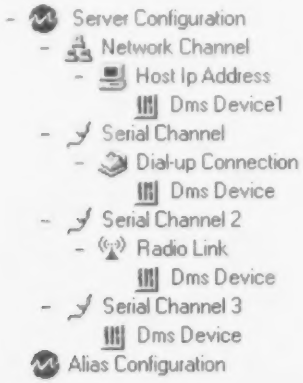

Figure 2-1. Sample Object Hierarchy

Once the object configuration is complete, a communication path to the end device is established and OPC client applications can connect to the OPC server and get the data from those end devices.

2.1.10 Common object communication requirements

2.1.10.1 All communication objects must be able to recover from a loss of communication

2.1.11 Communication object Statistics

2.1.11.1 All communication objects must support a sent count as an OPC Item

This enables OPC clients to monitor the number of messages that have been sent through this communication object. Because this is presented as a standard OPC data item any OPC client can get access to this information.

2.1.11.2 All communication objects must support a received count as an OPC Item

This enables OPC clients to monitor the number of messages that have been received back to this communication object. Because this is presented as a standard OPC data item any OPC client can get access to this information.

2.1.11.3 All communication objects must support a Timed out count as an OPC Item

This item provides a count for the number of times this communication object has time out on receiving a response when attempting to send a message. Because this is presented as a standard OPC data item any OPC client can get access to this information.

2.1.11.4 All communication objects must support a Retried count as an OPC Item

This enables OPC clients to monitor the number of times a communication object has attempted to re-send it communication. Because this is presented as a standard OPC data item any OPC client can get access to this information.

2.1.11.5 All communication objects must support a failed count as an OPC Item

This enables OPC clients to monitor the number of times a communication object has gone through its reconnection algorithm and eventually failed to communicate. Because this is presented as a standard OPC data item any OPC client can get access to this information.

2.1.11.6 All communication objects must support a overrun count as an OPC Item

Total number of times that a scheduled poll has come due a second time before it has yet been sent and cleared from the due list. This type of information is only relevant in a SCADA type OPC server because of the way the OPC server manages field communication. Because this is presented as a standard OPC data item any OPC client can get access to this information.

2.1.12 Network Channel Requirements

2.1.12.1 A network channel object must be provided

The network channel controls the polling activity on the network.

2.1.12.2 The network channel object must a configurable inter-device delay parameter

The inter-device delay is the minimum amount of time (in milliseconds) between the last request/response to/from one unit and the next request to another unit on this channel.

2.1.12.3 The network channel object must a configurable reconnect delay parameter

The reconnect delay is the minimum amount of time (in milliseconds) to wait before attempting to reconnect after an attempt to connect to a channel has failed.

2.1.12.4 The network channel object must a configurable threshold rate parameter

The Threshold rate is the maximum achievable scan interval (in milliseconds) for the circuit. The server polls items with update rates faster than this on a continuous basis.

2.1.13 Host Connection Requirements

2.1.13.1 A host connection object must be provided

This object is used to connect to a remote TCP/IP address on an Ethernet network.

2.1.13.2 The network channel object must have a configurable IP address

2.1.13.3 The network channel object must have a configurable port address

2.1.14 Serial Channel Requirements

2.1.14.1 Must support direct RS232 communication

2.1.14.2 It must conform to NTCIP 2101 when using an RS-232 link

2.1.14.3 The server must support a property for baud rate

2.1.14.4 The server must support a property for data bits

2.1.14.5 The server must support a property for parity

2.1.14.6 The server must support a property for stop bits

2.1.14.7 The server must support RTS/CTS flow control

2.1.14.8 The server must support DCD transmission collision avoidance with random back-off time

2.1.14.9 The server must allow the user to preset control line states (RTS, DTR)

This is to accommodate converters (RS232/RS485, RS232/RS485, etc)

2.1.15 Radio Link object Requirements

2.1.15.1 The Radio Link object must support properties for RTS key up and RTS key down

Key up/down delays are used for transceiver warm-up and cool down, with different delays for one or more repeater on the same channel.

2.1.16 Dial-up object Requirements

2.1.16.1 The server must support Hayes-compatible dial-up modems via serial connection

2.1.16.2 The server must support any modem that responds to the standard AT command set MNP10 and MNP5

A recommended modem for testing is the USR model 5686.

2.1.16.3 The server must support v90BIS

If we support this we will automatically support v34 and v42

2.1.16.4 The server must tolerate any character case and white space in command responses

2.1.16.5 The server must issue reset and initialization commands

This includes before dialing out *and* after hanging up

2.1.16.6 The server must support communication directly with CDPD

CDPD refers to Cellular Digital Packet Data modems using the Hayes standard AT command set.

2.1.16.7 The network channel must support Point-to-Point Protocol (PPP)

PPP is a peer-to-peer relationship. PPP has only 2 devices, called "peers".

2.1.16.8 The network channel must support Point-to-Multi-Point Protocol (PMPP)

The following is a brief description of the PMPP protocol:

"PMPP implies a primary/secondary relationship, with one primary device and several secondary devices. It is a connectionless delivery mechanism and does not support contention based communication. PMPP is primarily responsible for the Physical Layer (FSK or RS-232), and the Data Link Layer. The Data Link layer ensures correct transmission of data and incorporates mechanisms for ensuring the integrity of data, also acting to ensure that no data is lost if possible. Provides error detection, link activation and deactivation control, and notification. A secondary device will not transmit unless explicitly allowed to by a primary."

Only primary station requirements are implemented for the PMPP protocol. Secondary station emulation is not supported.

2.1.16.9 The OPC server must conform to NTCIP 2102 (PMPP/FSK) and NTCIP 2101(PMPP/RS-232)

The server may manage dial-up modems with data rates of 28,8 kbps, 19,2 kbps,14.4 kbps, 9600 bps, 4800 bps, 2400 bps, 1200 bps, 600 bps, and 300 bps. This requirement will be tested with as many of these rates as allowed by the chosen modem hardware selected for acceptance testing.

2.1.17 DMS Device object Requirements

2.1.17.1 A DMS Device object must have an associated MIB

DMS devices have MIB's that describes the NTCIP objects they support.

2.1.18 Configuration and User Interface

2.1.18.1 The OPC server must have a graphical user interface for configuration (GUI)

2.1.18.2 The user must be able to open an existing server configuration from an XML file

2.1.18.3 The user must be able to "save/save as" a server configuration to an XML file

2.1.18.4 The server must be able to automatically save the server configuration on shutdown

2.1.18.5 The server must be able to automatically load a server configuration on startup

2.1.18.6 The users must be able to configure the path to the device

2.1.18.7 The user must be able to create aliases and save them in the server's XML file

2.1.18.8 The user must be able to export the aliases as a csv file

2.1.18.9 The user must be able to import the aliases from a csv file

2.1.18.10 The GUI must be able to update the OPC server with online configuration changes

2.1.18.11 The GUI will provide a SCADA configuration interface for SCADA devices

2.1.18.12 The OPC server shall provide a way to perform some configuration activities remotely

This would include activities like adding new devices, changing the communication parameters, removing existing devices etc. Remote configuration is a requirement, even though no decision has been made on how to do this.

2.1.19 Logging

2.1.19.1 The server must support the creation of a log file when running.

2.1.19.2 The server must provide four levels of logging: none, low, medium, high

2.1.19.3 The server must allow the user to configure the name and location of the log file

2.1.19.4 The server must provide a filtering mechanism for log files

The server must allow the user to name and create a "filtered" log file containing lines that have a particular text string in them.

2.1.19.5 The server must be able to overwrite old logs automatically

2.1.19.6 The log file must increase to a maximum size of 10 MB and then wrap around

2.1.19.7 The server must provide the option of "committed writes"

This option guarantees no log statements are lost in a crash. This option, when selected, must produce a log file that grows indefinitely.

2.1.19.8 The server must allow the user to create "interface logs"

An interface log is a log file created for every client connection to the server.

2.1.19.9 The interface logs must support the following log levels: none, low, medium, high

2.1.19.10 The interface logs must support the option of "committed writes"

This guarantees no log statements are lost in a crash.

2.1.19.11 When using committed writes the server must produce a log file that grows indefinitely

2.1.20 NTCIP DMS Object Requirements

This section outlines requirements specific to the NTCIP definitions for the objects that DMS devices support.

2.1.20.1 The OPC server must provide support for proprietary DMS objects provided MIBs are provided by the proprietor of the proprietary DMS objects

Additional MIB's may be added to the existing set of standard MIB's as long as the additional MIB does not collide in the namespace.

2.1.21 DMS Property Management requirements

2.1.21.1 The OPC Server must provide single read or write access to DMS parameters

The DMS devices have a set of NTCIP object contained within them. However, in some cases writing (or changing these values) requires a sequence of steps to be executed rather than just one simple write access. These properties (like the property "Message" for example) must be presented within the OPC server interface as single OPC items. Reading and writing these simple OPC items

should not require the OPC client to perform any additional functions but instead the OPC server must abstract these complex sequences transparently to the OPC client.

2.1.21.2 The OPC server must manage multiple write requests from multiple clients

The OPC server is responsible for ensuring that there are no synchronization issues when multiple clients attempt to write to the same value within a device at the same time. This is especially important as performing a write operation may require a series of writes to a number of different objects and interruption of that sequence could cause faults in the communication.

2.2 Non-Functional Requirements

2.2.1 Software Interface Requirements

2.2.1.1 The OPC Server for NTCIP must be built using the Matrikon SCADA framework

Matrikon has 2 frameworks from which to build OPC servers. The first is considered the plant edition framework, and the second is the SCADA edition framework. The SCADA edition framework is better suited to this project as it manages communication to remote field devices using communications algorithms that are much more robust and therefore less susceptible to communication failures.

Note: Management of licensing for either of these frameworks is the same.

2.2.1.2 The OPC server must interoperate with the IDI Tester application

The stakeholders of this project have agreed that the Matrikon OPC server for NTCIP must interoperate with the IDI tester application. This application was provided by MTQ, and will be the main test of interoperability of the OPC server with NTCIP.

2.2.2 OPC Interfaces

2.2.3 Common Interfaces

2.2.3.1 The server must support the mandatory functionality specified in OPC Common Definitions and Interfaces

2.2.3.2 The server must allow the client to set the *client name* for each connection

2.2.3.3 The server must support the *shutdown* event notification client-side interface

2.2.3.4 The server must support the US English *locale* for result code translation

2.2.3.5 The server must support the System default *locale* for results code translation

2.2.3.6 The server must support the User default *locale* for results code translation

2.2.3.7 The server must support Neutral *locale* for results code translation

2.2.3.8 OPC Data Access Interfaces

2.2.3.9 The server must support all mandatory functionality defined by the OPC Data Access Custom Interface Standard

2.2.3.10 The server must support OPC DA version 1.0a, 2.0x, and 3.0

2.2.4 Communication Interface Requirements

2.2.4.1 The OPC server must use NTCIP standards for communication to the DMS devices

2.2.4.2 The OPC server must be able to manage multiple communication paths simultaneously

In other words the OPC server must be able to not only communicate with multiple devices using NTCIP, but it must also attempt to optimize that communication should it be running multiple devices over a single communication link (serial network).

2.2.4.3 The OPC server must be able to recover from lost communication

2.2.4.4 The OPC server must be able to mange noisy communication

For example, communication through a radio link can be very noisy and produce a lot of retransmission. As stated in the various object requirements, some configuration parameters are provided to allow the end user the ability to optimize the communication depending on the quality of the communication present.

2.2.4.5 The OPC server shall conform to NTCIP 2301 as a Managed Agent

2.2.4.6 The OPC server shall be meet Conformance Level 1 (SNMP)

2.2.4.7 Responses must use the same Application Profile used by the request

2.2.4.8 The NTCIP server shall support the receipt of Application data packets

This means the must be allowed at any time as specified by the subject standards

2.2.4.9 The OPC server must support the use of TCP/IP communication

As specified by NTCIP 2202.

2.2.4.10 The OPC server may support the use of UDP communication as specified by NTCIP 2202

2.2.4.11 If supporting UDP, response datagram's must use the same Transport Profile used in the request

2.2.4.12 The server shall also conform to NTCIP2201 for the T2-NULL protocol

2.2.4.13 The OPC server must allow for up to 230.4 kbps when on a fibre optic network

In other words, the OPC server can be used to communicate over any Ethernet networks so it must be designed in a way that it does not constrain is communication speeds in any way. Therefore, theoretically if we increase the bandwidth or speed of the underlying network the communication for the OPC server should increase as well. It is noted here that the OPC server itself will not be able to tell what type of network it is communicating over.

2.2.5 Documentation Requirements

2.2.5.1 Matrikon must provide a third party certificate of conformance for NTCIP

This will be an agreed upon third party such as Trevilon or other recognized leading expert in the field of certifying NTCIP components.

Test tools and products developed by Trevilon (www.trevilon.com) and Frontline Test Equipment (www.fte.com) shall be considered in the conformance review, as they have been involved with highly reputable organizations and their products are recognized by the ITS / NTCIP community.

Furthermore, site (field) testing of the OPC server shall be considered given the availability of NTCIP DMS signs (Daktronics) signs procured by the Ministère des transports du Québec.

2.2.5.2 A test plan shall be developed following the completion of conformance review conducted by Matrikon.

2.2.5.3 Must provide a document for the complete MIB

This document will be used for any third parties in describing compliant devices.

2.2.5.4 Must provide the PICS protocol implementation table for each NTCIP standard supported

3 REQUIREMENTS TABLE AND TRACEABILITY

This table provides a mapping from the requirements defined within this document to a static requirement identification number. The identification numbers can be used throughout the project to reference the requirements.

Note: This table will be completed when the SRD just prior to sign off by all parties.

Table 1. Requirements Cross Reference Matrix

Req. ID	Section Reference	Description
001	2.1.1.1	The server must support all Global Objects
002	2.1.1.2	The server must support all DMS Object Definitions
003	2.1.1.3	The server must support the NTCIP Application Level protocol SNMP
004	2.1.1.4	The server must support the NTCIP Transport Level protocol T2/NULL
005	2.1.1.5	The server must support the NTCIP Transport Level protocol TCP/IP
006	2.1.1.6	The server must support the NTCIP Subnetwork Level specifications for PPP over RS-232
008	2.1.1.7	The server must support the NTCIP Subnetwork Level specifications for PMPP over RS-232
009	2.1.1.8	The server must support the NTCIP Subnetwork Level specifications for PMPP using FSK Modems
010	2.1.2.1	The server must run as an out-of-process COM server
011	2.1.2.2	The server must run optionally as an NT/2000 service or as a regular process
012	2.1.2.3	The server must use /SERVICE as the command line parameter to toggle registration
013	2.1.2.4	The server must provide a command line option for registration and deregistration
014	2.1.2.5	The server /REGSERVER as the command-line option for registration
015	2.1.2.6	The server /UNREGSERVER as the command-line option for deregistering
016	2.1.2.7	Server must register the OPC sub-key and vendor information string
017	2.1.2.8	Server must be assigned a globally unique application ID number

NETSNMP COPYING AGREEMENT

Various copyrights apply to this package, listed in 4 separate parts below. Please make sure that you read all the parts. Up until 2001, the project was based at UC Davis, and the first part covers all code written during this time. From 2001 onwards, the project has been based at SourceForge, and Networks Associates Technology, Inc hold the copyright on behalf of the wider Net-SNMP community, covering all derivative work done since then. An additional copyright section has been added as Part 3 below also under a BSD license for the work contributed by Cambridge Broadband Ltd. to the project since 2001. An additional copyright section has been added as Part 4 below also under a BSD license for the work contributed by Sun Microsystems, Inc. to the project since 2003.

Code has been contributed to this project by many people over the years it has been in development, and a full list of contributors can be found in the README file under the THANKS section.

---- Part 1: CMU/UCD copyright notice: (BSD like) -----

Copyright 1989, 1991, 1992 by Carnegie Mellon University

Derivative Work - 1996, 1998-2000
Copyright 1996, 1998-2000 The Regents of the University of California

All Rights Reserved

Permission to use, copy, modify and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appears in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of CMU and The Regents of the University of California not be used in advertising or publicity pertaining to distribution of the software without specific written permission.

CMU AND THE REGENTS OF THE UNIVERSITY OF CALIFORNIA DISCLAIM ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL CMU OR THE REGENTS OF THE UNIVERSITY OF CALIFORNIA BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM THE LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

---- Part 2: Networks Associates Technology, Inc copyright notice (BSD) -----

Copyright (c) 2001-2003, Networks Associates Technology, Inc
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the Networks Associates Technology, Inc nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

---- Part 3: Cambridge Broadband Ltd. copyright notice (BSD) -----

Portions of this code are copyright (c) 2001-2003, Cambridge Broadband Ltd. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* The name of Cambridge Broadband Ltd. may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDER ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

---- Part 4: Sun Microsystems, Inc. copyright notice (BSD) -----

Copyright © 2003 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054, U.S.A. All rights reserved.

Use is subject to license terms below.

This distribution may include materials developed by third parties.

Sun, Sun Microsystems, the Sun logo and Solaris are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the Sun Microsystems, Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

---- Part 5: Sparta, Inc copyright notice (BSD) -----

Copyright (c) 2003-2004, Sparta, Inc
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the Networks Associates Technology, Inc nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

APPENDIX B-2
SPECIFIC REQUIREMENTS - COMMENTS

OPC Server for NTCIP

2004Sep23

DDE-TCS4502

1 COMMENTS

This document contains a set of comments provided to Matrikon in reference to the specific requirements document version 1.2 for the OPC server for NTCIP. This document also contains the responses for each of these comments. A new version of the SRD will be issued to clarify the comments in this document.

1.1 General

Comment

The requirements set out in the document appear to be standardized: Matrikon has established a standard process to develop OPC servers and this process is being applied to the development of a NTCIP / OPC server. In fact, the envisaged server is simple and acts primarily as a data translator. The OPC client application (traffic management and /or VMS management, software) may have to manage some tasks normally executed at a communications server level.

Response

We considered this at one point during the development of the requirements for the OPC server and had decided that it would be a more useful product if the various NTCIP objects were abstracted to a degree where the OPC client application would not need to be specially written in order to manage simple operations. For example, writing to a message on a DMS sign should really just be a simple write operation to a string tag value (called an OPC Item in OPC) called "Message". The process of performing that write should be hidden from the client.

That said, we are considering also exposing the objects directly as well. This is really a balancing between simplicity and flexibility. The more simple the interface becomes, potentially the less flexible it could be. So if we provide a way to access these objects that is simple, but also provide a "back door" approach that an OPC client could use should they require more advanced functionality then this would be desirable. However, the advanced approach is considered an additional feature and may not be implemented in the final product should we find this to either be un-useful or if project constraints prevent us from implementation.

1.2 Standard MIB's (ref 2.1.1)

Comment

It is stated that the OPC server will support all standard MIBs; this means that all compulsory and optional MIBs are being supported. Please confirm.

Response

The server will support all MIB's provided by NTCIP. They may be slightly modified to remove white space and comments if there are issues with the MIB compiler. Whether these objects are supported by the DMS is an important question to ask. If an object does not exist on the other end, then the object will show up as bad quality. However note that as stated in section

1.1, when we are doing a translation for the OPC client we may just be showing bad quality on DMS properties if these properties are incorrect or do not match the actual device.

1.3 Proprietary DMS objects (ref 2.1.20)

Comment

Consider adding a more general statement to the effect that "OPC Server must provide support for proprietary DMS objects provided MIBs are provided by the proprietor of the proprietary DMS objects".

Response

We have decided to update the SRD to reflect this comment.

1.4 Multiple OPC Clients

Comment

It is not clear how, or if the OPC server will handle any DMS management issues with multiple OPC clients. For example, requesting an immediate message to be displayed on a sign requires writing of multiple objects, in a specific order. Will the OPC server manage multiple OPC clients and properly mitigate the write operations or will only one OPC client be allowed to write at once and the OPC client application will thread the requests.

Response

Because the OPC server manages all write operations on behalf of the OPC client, the OPC server has control over these operations. Thus, if multiple clients are attempting to write to the same DMS property, the OPC server must complete the first client's write request before beginning the second. The OPC server also queues these requests sequentially as they are received by the clients.

1.5 Testing (ref 1.2.1.4)

Comment

We ask that the test tools and products developed by Trevillon (www.trevilon.com) and Frontline Test Equipment (www.fte.com) be considered in your review, as they have been involved with highly reputable organizations and their prioducts are recognized by the ITS / NTCIP community.

Furthermore, site (field) testing of the OPC server is to be considered given the availablility of NTCIP VMS signs (Daktronics) signs procured by the Ministère des transports du Québec.

We expect that a test plan will be developed following the completion of "test tools" review conducted by Matrikon.

Response

We will use one of these vendors as our third party company test company. Please note that the MTQ has instructed us to use the IDI tester application provided by MTQ. They have requested that we use this software application as the official NTCIP simulator and that the OPC server must be interoperable with this application before the final product can be accepted. I think it would be wise for Matrikon to use the Trevilon or FTE software if we decided to use them as our third party test company since this would reduce the risk of failing any of their testing.

1.6 Matrikon SCADA Framework

Comment

Clause 2.2.1.1 - states that the OPC Server must be built using the Matrikon SCADA framework. Does this mean the OPC Server will be dependant on another product and as such, to use the OPC Server, a license to the Matrikon SCADA framework is required, or will the OPC Server be an independent product?

Response

Matrikon has two frameworks they can use during the development of any new OPC server interface. One is designed for plants, the other for SCADA systems. Plant edition OPC servers are the more common and are designed to have good, dependable communication with lots of bandwidth, the typical case for a LAN environment. The SCADA framework is designed for SCADA systems that can have poor communication over unreliable connections like radio, dial-up, etc. The SCADA framework is really an enhanced plant framework as it supports everything the plant framework can do and more. Externally the OPC servers built with either of these frameworks appear identical except for the additional configuration capabilities of the SCADA server (adding modems, radios, etc). Due to the requirements of this project the SCADA framework would be a better fit. Finally, there are no differences in licensing for this product.

APPENDIX C
RAPPORT D'ESSAIS SUR SITE

Transports Québec

SERVEUR DE COMMUNICATION OPC POUR NTCIP

MANDAT NO° : 5230-04-QZ02

RAPPORT D'ESSAIS SUR SITE DU SERVEUR DE COMMUNICATION OPC POUR NTCIP

Janvier 2005

Révision 01

Tecsult ref :° :05-12757

TECSULT

TABLES DES MATIÈRES

1. INTRODUCTION 1

2. MISE EN SITUATION 1

3. RÉFERENCE 2

3.1 Acronymes 2

3.2 Références 3

4. ÉTENDUE DES ESSAIS 4

4.1 Méthodologie 4

4.1.1 Essais à distance 4
4.1.2 Essais sur site 4

4.2 Sommaire des essais 5

5. RETOUR SUR LES ESSAIS 6

6. CONCLUSION ET RECOMMENDATIONS 7

ANNEXE C-1 – Tableaux des résultats des essais

Liste des tableaux

Tableau 1 Liste des acronymes 2
Tableau 2 Dictionnaires de données et protocoles de communications 3

1. INTRODUCTION

Ce document a pour but de présenter les essais sur site qui ont été réalisés sur le serveur de communication OPC pour NTCIP. Les procédures d'essais sont décrites dans les paragraphes qui suivent. Les essais avaient pour but de valider l'interopérabilité du serveur de communication avec les contrôleurs de panneaux à messages variables de deux (2) manufacturiers dont les PMV sont utilisés par le Ministère.

Ces essais sur site devaient permettre de faire l'acceptation provisoire du serveur de communication tel que conçu et présenté par Matrikon.

2. MISE EN SITUATION

Tel qu'élaboré en début de projet, le serveur OPC pour NTCIP avait pour but de permettre l'interopérabilité d'une interface de contrôle unique (client OPC) avec les panneaux à message variables de différents manufacturiers. L'étendue des travaux consistait à la conception du serveur de communication uniquement et ne comprenait pas le développement d'une interface de contrôle unique mais permettra celui-ci.

Les restrictions escomptées résidaient dans le niveau de conformité des panneaux à messages variables des différents manufacturiers. En effet, étant donnée la faible maturité du standard NTCIP et l'absence d'un organisme de certification, les standards laissent parfois place à interprétation.

Afin de minimiser l'absence d'uniformisation des différents manufacturiers dans l'implémentation des standards NCTIP, une référence a été établie dans le cadre du projet de conception du serveur de communication OPC pour NTCIP. La référence adoptée est le logiciel «NTCIP tester» de la firme Intelligent Devices Inc.

En résumé, le mandat consiste donc en la conception d'un serveur de communication OPC pour NTCIP supportant certains profiles de communication et supportant un ensemble d'objets NTCIP. Ceux-ci sont tels qu'il l'est défini dans le document de conception «OPC Server for NTCIP, Specific Requirements 1.4» et dont l'implémentation doit respecter la référence donnée, soit le logiciel «NTCIP tester».

3. RÉFERENCE

3.1 Acronymes

Tableau 1 Liste des acronymes

Acronymes	Définition
ASC	Actuated traffic Signal Controller
CDPD	Cellular Digital Packet Data
CCTV	Closed Circuit Television
DCM	Data Collection and Monitoring Devices
DMS	Dynamic Message Sign
ESS	Environmental sensor stations
FSK	Frequency Shift Key
IP	Internet Protocol
ITS	Intelligent Transportation Systems
MIB	Management Information Base
NEMA	National Electrical Manufacturers Association
NTCIP	National Transportation Communications for ITS Protocol
OPC	OLE for Process Control
PICS	Protocol Implementation Conformance Statement
PMPP	Point-to-Multi-Point Protocol
PPP	Point-to-Point Protocol
SNMP	Simple Network Management Protocol
UDP	User Datagram Protocol

3.2 Références

Ce document fait référence à plusieurs standards selon leur désignation NTCIP. La liste suivante donne une référence complète, incluant la version exacte, des standards utilisés pour la conception du serveur de communication.

Tableau 2 Dictionnaires de données et protocoles de communications

Référence NTCIP	Numéro du document	Titre
1101	NEMA TS 3.2-1996	*STMF : Simple Transportation Management Framework*
1102	OER	Octet encoding Rules
1103	STMP	Transportation Management Protocol
1201	NEMA TS 3.4-1996	*Global Object Definitions*
1203	NEMA TS 3.6-1997	*Object Definitions for Dynamic Message Signs*
2101	SP-PMPP/RS232	*Subnetwork Profile for Point to Multi-Point Protocol usng RS-232*
2102	SP-PMPP/FSK	*Subnetwork Profile for Point to Multi-Point Protocol using FSK modems*
2103	SP-PPP/RS232	*Subnetwork Profile for Point-to-Point Protocol using RS 232*
2201	TP-Null	Null Transport Profile
2202	TP-Internet	*Internet Transport Profile*
2301	AP-STMF	*Simple transportation Management Framework*

4. ÉTENDUE DES ESSAIS

4.1 Méthodologie

4.1.1 Essais à distance

De premiers essais à distance ont permis de valider certains aspects des profiles de communication. Ces essais, assez sommaires, visaient à minimiser l'incertitude sur la composante la plus risquée du projet soit l'implémentation des profiles de communication.

Les essais ont été réalisés sur un PMV mobile du manufacturier Ver-Mac situé au centre de signalisation du ministère des Transports à Québec. Le PMV mobile était configuré en mode NTCIP en utilisant le profile de communication SNMP/T2/PMPP/modem cellulaire. Les essais ont été réalisés les 29 et 30 novembre via téléphone par la firme Matrikon à partir de Edmonton. Le contrôleur du PMV était conforme à la version 1 du standard NTCIP pour le PMV soit NTCIP 1203 :1997.

Les essais sommaires effectués consistaient à :

1. Établir la communication selon le profile cité plus haut;
2. Valider la fiabilité de la connexion;
3. Valider la récupération d'une perte de communication;
4. Lire les objets de configuration NTCIP du PMV

Les essais ont été réalisés avec succès.

4.1.2 Essais sur site

De façon sommaire, le document de spécifications pour la conception du serveur de communication OPC pour NTCIP prévoit que le serveur de communication supporte tous les objets NTCIP pour les PMV. Les objets NTCIP obligatoires et optionnels supportés sont tous les objets prévus aux standards :

- NTCIP 1203 :1997; (dictionnaire d'objets pour les PMV)
- NTCIP 1201 :1996. (dictionnaire d'objets globaux)

En plus, le document de spécifications prévoit le support de certains standards et profiles de communication tel qu'indiqué par les standards NTCIP suivants:

- NTCIP 2301 (SNMP)
- NTCIP 2202 (TCP et IP)
- NTCIP 2201 (T2)
- NTCIP 2101 (PMPP sur RS232)
- NTCIP 2103 (PPP sur RS232)

Les essais visaient à valider le support de tous les objets NTCIP pour PMV. Les essais permettaient aussi de valider si certains objets définis par le serveur de communication étaient supportés par le contrôleur de PMV.

En effet, le serveur expose tous les objets du MIB standard prévus par NTCIP :1203 et NTCIP 1201. Le MIB est la base de données de définition des objets qui peuvent être gérés par SNMP. Si un objet n'est pas supporté par un contrôleur de PMV, un message d'erreur est retourné au serveur lui indiquant ce fait. Les essais possibles sur un PMV particulier sont donc restreints au support de ces objets par le contrôleur du PMV.

Les essais ont eu lieu du 13 au 17 décembre. Les essais ont été réalisés dans les configurations suivantes pour assurer une interopérabilité du serveur de communication avec les contrôleurs de PMV de différents manufacturiers.

1. PMV mobile Ver-Mac avec SNMP/T2/PMPP/téléphone cellulaire
2. PMV mobile Ver-Mac avec SNMP/T2/PMPP/RS232 direct
3. PMV fixe Daktronics avec SNMP/T2/PMPP/RS232/fibre optique

Le banc d'essais consistait en un ordinateur portable sur lequel le représentant de Matrikon a fait l'installation du serveur OPC pour NTCIP. Aussi, le logiciel client OPC « OPC Explorer » de Matrikon a été utilisé. Il est à noter que d'autres logiciels client OPC auraient permis de réaliser les mêmes essais. Le client OPC n'est utilisé que pour présenter les objets NTCIP sous forme de liste et pour donner une interface pour modifier ces variables du contrôleur PMV.

Les essais ont eu lieu au centre de service Anjou du ministère des Transports pour les configurations 1 et 2 énumérées plus haut. Les essais pour la configuration 3 ont eu lieu à partir du CSGC. M.Jean Morin représentant du ministère des Transports a assisté à une démonstration des fonctionnalités pour la configuration 1 (complet) et 3 (partiellement).

4.2 Sommaire des essais

Les tableaux d'essais sont présentés en annexe. Les tableaux liste la totalité des objets obligatoires et optionnels des standards 1203 et 1201 et qui ont été testés.

Les essais sur les objets pour la configuration 1 et 2 sont traités dans le même tableau puisqu'il s'agit évidemment du même contrôleur et que les résultats sont les mêmes.

Pour chacun des objets, le tableau indique la valeur lue au contrôleur de PMV. Dès le moment où on peut lire une valeur dont l'exactitude peut être vérifiée et écrire la valeur des objets ayant une propriété d'écriture, on prouve le support de ces objets par le serveur OPC.

En surplus, lorsqu'on arrive à activer une fonctionnalité du PMV, on démontre la compréhension de l'implémentation du standard par le manufacturier, ce qui en soit n'est pas l'objectif premier visé par ces essais.

Pour l'ensemble des objets, des essais sur les fonctionnalités ont été réalisés. Sur le PMV Vermac, les essais principalement sur les groupes de conformité «Pixel service» et «Time base event schedule» n'ont pu être validés faute de temps. Sur le panneau Daktronics en service sur le réseau routier, les essais fonctionnels ont été plutôt limités. Cependant nous retenons que les fonctionnalités des mêmes groupes de conformité n'ont pu être validés. Dans les deux cas, il a

été néanmoins prouvé que les objets étaient disponibles dans le serveur de communication et sur le contrôleur de PMV. Plus de temps et d'information auprès des manufacturiers seraient nécessaire pour arriver à activer les fonctionnalités.

5. RETOUR SUR LES ESSAIS

Les essais effectués doivent être interprétés avec un certain recul. En effet, cette première période d'essais du serveur de communication vise à s'assurer que le serveur a la capacité de lire et écrire les objets NTCIP des contrôleurs des 2 manufacturiers. Du même coup, ils permettent aussi de faire un certain apprentissage sur la façon dont chacun des manufacturiers a implémenté ces objets NTCIP.

Bien que le serveur puisse lire et écrire un objet au contrôleur de PMV, il faut que le PMV supporte la fonction désirée. De même, afin de réaliser une fonction sur un PMV, l'utilisateur du serveur doit comprendre la séquence d'objets à écrire, mais encore plus la façon dont le contrôleur du manufacturier s'attend à recevoir ces données. Pour la grande majorité des objets, les essais ont permis de valider que le standard était assez précis et que les 2 manufacturiers ont implémenté les fonctionnalités de la même façon. Cependant, dans certains cas et possiblement puisque le standard laissait place à interprétation, la fonctionnalité n'a pu être vérifiée chez l'un ou l'autre des manufacturiers. Un travail d'apprentissage et de consultation avec les manufacturiers sera nécessaire pour clarifier ces points. Il semble à priori que ces quelques points de vérification ne soient pas très problématiques et pourront se clarifier après discussion avec les manufacturiers.

Cependant, une différence majeure et très importante fut notée en comparant l'implémentation des fonctionnalités des 2 manufacturiers. Cette différence se situe dans la façon d'activer un message sur un PMV. Cette fonction requiert entre autres, le paramètre « dmsMessageCRC ». Ce paramètre de l'objet de contrôle NTCIP « dmsActivateMessage » permet d'activer un message lorsque concaténer avec d'autres.

L'objet « dmsMessageCRC » est de format «INTEGER», il est donc composé de 2 bytes d'information. Les contrôleurs de PMV des 2 manufacturiers s'attendent à lire ces deux bytes d'information dans un ordre particulier. Les 2 manufacturiers en question ont inversé, l'un par rapport à l'autre, l'ordre dans lequel le contrôleur s'attend de recevoir ces informations. En résumé, le serveur, ayant pour but de ne pas différentier le panneau avec lequel il communique (interopérabilité et interchangeabilité), envoie nécessairement cette information dans un ordre précis. Ceci implique que pour les panneaux d'un des manufacturiers qu'il n'est présentement pas possible d'activer un message.

Ceci étant dit, pour les besoins des essais, une modification au serveur a été réalisée afin de permettre de compléter. Cette question est un exemple parfait démontrant que le serveur est fonctionnel mais que dû à une différence d'implémentation du standard par les 2 manufacturiers, il n'est pas possible à priori d'activer un message de la même façon sur les PMV sans devoir les différentier. La différentiation étant à l'opposé de ce qui est voulu du serveur de communication unique avec une interface unique, ce problème devra être discuté avec les manufacturiers afin d'établir la façon dont la fonction doit être implémentée.

6. CONCLUSION ET RECOMMENDATIONS

En conclusion, les essais réalisés furent concluants. Il a été démontré que tous les objets NTCIP pour les PMV, soient les objets des standard NTCIP1203 et NTCIP1201, sont supportés par le serveur de communication OPC pour NTCIP. Le serveur ainsi conçu permettra le développement et l'utilisation d'une interface unique pour contrôler les PMV de différents manufacturiers.

Au moment des essais, aucun objet propriétaire du Ministère n'avait été défini donc intégré au serveur de communication. Il est à considérer aussi que les objets propriétaires des manufacturiers ne sont pas supportés par le serveur de communication, le but étant d'uniformiser et de standardiser les mécanismes de communication avec les différents PMV peu importe le manufacturier. Si certains objets et certaines fonctionnalités devaient être retenus, il est recommandé que le Ministère les intègre à titre d'objets propriétaires ministériels que les manufacturiers pourront ou devront supporter.

En niveau des profiles de communication, les standards TCP et IP n'ont pas été validés sur site. Cependant, la firme Matrikon confirme leur fonctionnalité puisque qu'ils ont été validés au bureau. Au moment des essais, le standard PPP n'avait pas été implémenté dans le serveur. Ces travaux devront être réalisés en janvier 2005.

Il est recommandé que le Ministère amorce immédiatement une discussion avec les manufacturiers afin d'éclaircir les fonctionnalités qui n'ont put être réalisées sur les PMV. Principalement, il sera impératif de statuer sur la façon d'utiliser les objets « dmsMessageCRC » avec « dmsActivateMessage » et d'uniformiser leur utilisation.

Il est recommandé aussi de préparer un banc d'essais final pour valider quelques objets NTCIP qui méritent une deuxième vérification, mais surtout pour tester les standards de communication non validés soit PPP et TCP/IP. C'est essais devront avoir lieu immédiatement au début de février 2005. Ces essais finaux devront être supervisés par un ou des représentants du Ministère afin de faire l'acceptation finale du serveur de communication.

ANNEXE C-1

Tableaux des résultats des essais

Légende	➳	Indique que l'objet NTCIP est existant dans le contrôleur PMV, il est vu par le serveur de communication et que la fonctionnalité a été vérifiée
	!	Indique que l'objet NTCIP est existant dans le contrôleur PMV, il est vu par le serveur de communication mais que la fonctionnalité n'a pu être vérifié. Pour les besoins de validation du serveur de communication ceci indique que le serveur supporte l'objet cependant le manufacturier ne supporte pas la fonctionnalité ou la fonctionnalité n'a pu être testée.
	X	Indique que le serveur de communication n'arrive pas à lire cet objet du contrôleur de PMV. Pris seul ce test pourrait indiquer que le serveur de communication ou le contrôleur de PMV ne supporte pas l'objet. Cependant, en le prenant avec les essais sur le contrôleur du 2ieme manufacturier on peut conclure que l'objet est supporté par le serveur de communication, cependant il ne l'est pas par le contrôleur de PMV.

Groupe de conformité/ Objets NTCIP	Objet supporté	Valeur de l'objet	Commentaire
1203 :Sign Configuration and Capability			
dmsSignType	➳	6	Signification: full matrix
dmsBeacontype	➳	2	Signification: none
1203:GUI Appearance			
dmsSignAccess	➳	2	Signification: walk-in
dmsSignHeight	➳	2376	mm
dmsSignWidth	➳	9240	mm
dmsHorizontalBorder	➳	300	mm
dmsVerticalBorder	➳	300	mm
dmsLegend	➳	2	Signification: Aucun
dmsSignTechnology	➳	2	Signification:LED
1203:VMS Configuration			
vmsCharacterHeightPixels	➳	0	Signification: Variable (pleine matrice)
vmsCharacterWidthPixels	➳	0	Signification: Variable (pleine matrice)
vmsSignHeightPixels	➳	34	pixels
vmsSignWidthPixels	➳	140	pixels
vmsHorizontalPitch	➳	66	mm centre à centre pixel

vmsVerticalPitch	➳	66	mm centre à centre pixel
1203:Font Definition			
numfonts	➳	8	
fontTable	➳		non accessible
fontIndex	➳		non accessible
fontNumber	➳	1	
fontName	➳	MTQ07x05	
fontHeight	➳	12	
fontCharspacing	➳	1	
fontLineSpacing	➳	0	
fontVersionID	➳	50246	
maxFontCharacters	➳	255	
characterTable	➳		non accessible
fontIndex	➳		non accessible
characterNumber	➳	48	
characterWidth	➳	5	
characterBitmap	➳		valeur: série de bits formant la définition d'un "0"
1203:Multi Configuration			
defaultBackgroundColor	➳	0	Signification: Black
defaultForegroundColor	➳	9	Signification: Amber
defaultFlashOn	➳	3	dixième de secs
defaultFlashOff	➳	3	dixième de secs
defaultFont	➳	1	
defaultJustificationLine	➳	3	Signification: center
defaultJustificationPage	➳	2	Signification: top
defaultPageOnTime	➳	30	dixième de secs
defaultPageOffTime	➳	0	dixième de secs
defaultCharacterSet	➳	2	Signification: eightbit
1203:Message Table			
dmsNumPermanentMsg	➳	3	
dmsNumChangeableMsg	➳	1	1 Message courrant chargé par le CGC
dmsMaxChangeableMsg	➳	500	
dmsFreeChangeableMemory	➳	998000	

dmsNumVolatileMsg	➳	0	
dmsMaxVolatileMsg	➳	0	
dmsFreeVolatileMemory	➳	0	
dmsMessageTable	➳		non accessible
dmsMessageMemoryType	➳		Donne les propriétés du message sélectionné selon les indexes utilisés. dmsMessageMultistring.3.1 par exemple est le code message du message 1 de la banque mémoire changeable (type3). La valeur de cet objet ce lit: "[pt25o0][jp3][jl3][fo1][cf9][cb0]AUT. DÉCARIE [nl0]CIRCULATION FLUIDE[nl0]MAXIMUM 70 KM/H[np][pt25o0][jp3][jl3][fo1]A-20 OUEST[nl0]CIRCULATION FLUIDE".
dmsMessageNumber	➳		
dmsMessageMultistring	➳		
dmsMessageOwner	➳		
dmsMessageCRC	➳		
dmsMessageBeacon	➳		
dmsMessagePixelService	➳		
dmsMessageRunTimePriority	➳		
dmsMessageMsgStatus	➳		
dmsValidateMessageError	➳	2	
1203:Sign Control			
dmsControlMode	➳	4	Signification: central
dmsSWReset	➳	0	
dmsActivateMessage	➳	MessAct	String de format "messageActivationCode"
dmsMessageTimeRemaining	➳	65535	Signification: Infinite duration
dmsMsgTableSource	➳	MessID	String de format "message IDcode": mem type7, mem no255, CRC 0 (message noir affiché)
dmsMsgRequesterID	➳	127.0.0.1	
dmsMsgSourceMode	➳	8	Signification: central
dmsMemoryMgmt	➳	2	Signification: normal
dmsActivateMsgError	➳	2	Signification: None. Certains essais ont permis de valider les codes d'erreurs 7 et 8
1203:Default Message			
dmsShortPowerRecoveryMessage	➳	MessID	MessageIDCode (type7,MessNo1,CRC=0) =>NOIR
dmsLongPowerRecoveryMessage	➳	MessID	MessageIDCode (type7,MessNo1,CRC=0) =>NOIR
dmsShortpowerLossTime	➳	60	
dmsResetMessage	➳	MessID	MessageIDCode (type7,MessNo1,CRC=0) =>NOIR
dmsCommunicationLossMessage	➳	MessID	MessageIDCode (type7,MessNo1,CRC=0) =>NOIR
dmsTimeCommLoss	➳	0	
dmsPowerLossMessage	➳	MessID	MessageIDCode (type7,MessNo1,CRC=0) =>NOIR
dmsEndDurationMessage	➳	MessID	MessageIDCode (type7,MessNo1,CRC=0) =>NOIR
1203:Pixel Service			
vmsPixelServiceDuration	!		Objets NTCIP existants dans le contrôleur PMV mais la fonctionnalité n'a pu être activé
vmsPixelServiceFrequency	!		
vmsPixelServiceTime	!		

1203:Multi Error			
dmsValidateMessageError	➳	2	Signification: None
dmsMultiSyntaxError	➳	2	Signification: None
dmsMultiSyntaxErrorPosition	➳	0	
dmsMultiOtherErrorDescription	➳	0	
1203:Illumination / Brightness			
dmsIllumControl	➳	2	Signification: photocell
dmsIllumMaxPhotocellLevel	➳	16000	
dmsIllumPhotocellLevelStatus	➳	8162	
dmsIllumNumBrightLevels	➳	255	
dmsIllumBrightLevelStatus	➳	170	
dmsIllumManLevel	➳	1	
dmsIllumBrightnessValues	➳		Objet NTCIP disponible, fonctionnalité non testé
dmsIllumBrightnessValuesError	➳	2	
dmsIllumLightOutputStatus	➳	37265	
1203:Auxiliary IO			
MaxAuxDigital	➳	112	
MaxAuxAnalog	➳	24	
auxIOTable			
auxPortType			Non validé
auxPortNumber			Non validé
auxDescription			Non validé
auxResolution			Non validé
auxValue			Non validé
auxPortDirection			Non validé
1203:Sign Status			
statMultiFieldRows	➳	11	
statMultiFieldTable	➳		non accessible
statMultiFieldIndex	➳		non accessible
statMultiFieldCode	➳	1	
statMultiCurrentFieldValue	➳	03:17	
statCurrentSpeed	➳	0	
statCurrentSpeedLimit	➳	0	
watchdogFailureCount	➳	2	
dmsStatDoorOpen	➳	0	
shortErrorStatus	➳	0	
controllerErrorStatus	➳	0	

1203:Pixel Error Status			
pixelFailureTableNumRows	➳	0	
pixelFailureTable	➳		non accessible
pixelFailureDetectionType	➳	0	
pixelFailureIndex	➳		non accessible
pixelFailureXLocation	➳	0	
pixelFailureYLocation	➳	0	
pixelFailureStatus	➳	0	
pixelTestActivation	➳	2	
1203:Lamp Error			
LampFailureStuckOn	X		
LampFailureStuckOff	X		
LampTestActivation	➳	2	
1203:Fan Error			
fanfailures	➳	0	
fanTestActivation	➳	2	
1203:Power Status			
signvolts	➳	0	Aurrait probablement du donner une valeur
lowFuelThreshold	➳	0	
fuelLevel	➳	0	
engineRPM	➳	0	
lineVolts	➳	0	Aurrait probablement du donner une valeur
powerSource	➳	4	Signification: AC line
1203:Temperature Status			
tempMinCtrlCabinet	➳	0	deg C
tempMaxCtrlCabinet	➳	0	deg C
tempMinAmbiant	➳	-6	deg C
tempMaxAmbiant	➳	-6	deg C
tempMinSignHousing	➳	2	deg C
tempMaxSignHousing	➳	2	deg C
1201:Global Configuration			
globalSetIDParameter	➳	0	
globalMaxModules	➳	8	
globalModuleTable	➳		non accessible

moduleNumber	➳	1	
moduleDeviceNode	➳		valeur: 1.3.6.1.4.1.1206.4.2.3
moduleMake	➳		valeur: Daktronics Inc.
moduleModel	➳		valeur: ED-11281
moduleVersion	➳		valeur: 3
moduleType	➳		valeur: 3.2.7.21
1201:Database Management			
dBCreationTransaction			N/A
dbVerifyStatus			N/A
dbVerifyError			N/A
1201:Time Management			
globalTime	➳		valeur: 1103301500
globalDaylightSaving	➳	3	
globalLocalTimedifferential	➳		valeur: 4294963696
1201:Time Base Event Schedule			
maxTimeBaseScheduleEntries	➳	0	
timebaseScheduleTable	➳		non accessible
timebaseScheduleNumber	➳	1	
timebaseScheduleMonth	➳	0	
timebaseScheduleDay	➳	0	
timebaseScheduleDate	➳	0	
timebaseScheduleDayPlan	➳	1	
maxDayPlans	➳	20	
maxDayPlanEvents	➳	12	
timebasedayPlanTable	➳		non accessible
dayPlanNumber	➳	0	
dayPlanEventNumber	➳	0	
dayPlanHour	➳	0	
dayPlanMinute	➳	0	
dayPlanActionNumberOID	➳		valeur: 13.6.1.4.1.1206.4.2.6.3.1.0
dayPlanStatus	➳	0	
1201:Report			
maxEventLogConfigs	➳	50	
eventLogConfigTable	➳		non accessible
eventConfigID	➳		objet existant mais fonctionnalité non testé
eventConfigClass	➳		objet existant mais fonctionnalité non testé
eventConfigMode	➳		objet existant mais fonctionnalité non testé
eventConfigCompareValue	➳		objet existant mais fonctionnalité non testé

eventConfigCompareValue2	➳		objet existant mais fonctionnalité non testé
eventConfigCompareOID	➳		objet existant mais fonctionnalité non testé
eventConfigLogOID	➳		objet existant mais fonctionnalité non testé
eventConfigAction	➳		objet existant mais fonctionnalité non testé
maxEventLogSize	➳		objet existant mais fonctionnalité non testé
eventLogTable	➳		objet existant mais fonctionnalité non testé
eventLogClass	➳		objet existant mais fonctionnalité non testé
eventLogNumber	➳		objet existant mais fonctionnalité non testé
eventLogID	➳		objet existant mais fonctionnalité non testé
eventLogTime	➳		objet existant mais fonctionnalité non testé
eventLogValue	➳		objet existant mais fonctionnalité non testé
maxEventClasses	➳		objet existant mais fonctionnalité non testé
eventClassTable	➳		objet existant mais fonctionnalité non testé
eventClassNumber	➳		objet existant mais fonctionnalité non testé
eventClassLimit	➳		objet existant mais fonctionnalité non testé
eventClassClearTime	➳		objet existant mais fonctionnalité non testé
eventClassDescription	➳		objet existant mais fonctionnalité non testé
eventClassNumRowsInLog	➳		objet existant mais fonctionnalité non testé
Security			
communityNameAdmin	➳	public ou admin	Les autres objets ne sont accessible qu'en utilisant le nom de communauté "admin"
communityNameMax	➳		
communityNameTable	➳		non accessible
communityNameIndex	➳		
communityNameUser	➳		
communityNameMask	➳		

Légende	➳	Indique que l'objet NTCIP est existant dans le contrôleur PMV, il est vu par le serveur de communication et que la fonctionnalité a été vérifiée
	!	Indique que l'objet NTCIP est existant dans le contrôleur PMV, il est vu par le serveur de communication mais que la fonctionnalité n'a pu être vérifié. Pour les besoins de validation du serveur de communication ceci indique que le serveur supporte l'objet cependant le manufacturier ne supporte pas la fonctionnalité ou la fonctionnalité n'a pu être testée.
	X	Indique que le serveur de communication n'arrive pas à lire cet objet du contrôleur de PMV. Pris seul ce test pourrait indiquer que le serveur de communication ou le contrôleur de PMV ne supporte pas l'objet. Cependant, en le prenant avec les essais sur le contrôleur du 2ieme manufacturier on peut conclure que l'objet est supporté par le serveur de communication, cependant il ne l'est pas par le contrôleur de PMV.

Groupe de conformité/ Objets NTCIP	Objet supporté	Valeur de l'objet	Commentaire
1203 :Sign Configuration and Capability			
dmsSignType	➳	6	Signification: full matrix
dmsBeacontype	➳	2	Signification: none
1203:GUI Appearance			
dmsSignAccess	➳	2	Signification: walk-in
dmsSignHeight	➳	2376	mm
dmsSignWidth	➳	9240	mm
dmsHorizontalBorder	➳	300	mm
dmsVerticalBorder	➳	300	mm
dmsLegend	➳	2	Signification: Aucun
dmsSignTechnology	➳	2	Signification:LED
1203:VMS Configuration			
vmsCharacterHeightPixels	➳	0	Signification: Variable (pleine matrice)
vmsCharacterWidthPixels	➳	0	Signification: Variable (pleine matrice)
vmsSignHeightPixels	➳	34	pixels
vmsSignWidthPixels	➳	140	pixels
vmsHorizontalPitch	➳	66	mm centre à centre pixel

vmsVerticalPitch	➤	66	mm centre à centre pixel
1203:Font Definition			
numfonts	➤	8	
fontTable	➤		non accessible
fontIndex	➤		non accessible
fontNumber	➤	1	
fontName	➤	MTQ07x05	
fontHeight	➤	12	
fontCharspacing	➤	1	
fontLineSpacing	➤	0	
fontVersionID	➤	50246	
maxFontCharacters	➤	255	
characterTable	➤		non accessible
fontIndex	➤		non accessible
characterNumber	➤	48	
characterWidth	➤	5	
characterBitmap	➤		valeur: série de bits formant la définition d'un "0"
1203:Multi Configuration			
defaultBackgroundColor	➤	0	Signification: Black
defaultForegroundColor	➤	9	Signification: Amber
defaultFlashOn	➤	3	dixième de secs
defaultFlashOff	➤	3	dixième de secs
defaultFont	➤	1	
defaultJustificationLine	➤	3	Signification: center
defaultJustificationPage	➤	2	Signification: top
defaultPageOnTime	➤	30	dixième de secs
defaultPageOffTime	➤	0	dixième de secs
defaultCharacterSet	➤	2	Signification: eightbit
1203:Message Table			
dmsNumPermanentMsg	➤	3	
dmsNumChangeableMsg	➤	1	1 Message courrant chargé par le CGC
dmsMaxChangeableMsg	➤	500	
dmsFreeChangeableMemory	➤	998000	

dmsNumVolatileMsg	➳	0	
dmsMaxVolatileMsg	➳	0	
dmsFreeVolatileMemory	➳	0	
dmsMessageTable	➳		non accessible
dmsMessageMemoryType	➳		Donne les propriétés du message sélectionné selon les indexes utilisés. dmsMessageMultistring.3.1 par exemple est le code message du message 1 de la banque mémoire changeable (type3). La valeur de cet objet ce lit: "[pt25o0][jp3][jl3][fo1][cf9][cb0]AUT. DÉCARIE [nl0]CIRCULATION FLUIDE[nl0]MAXIMUM 70 KM/H[np][pt25o0][jp3][jl3][fo1]A-20 OUEST[nl0]CIRCULATION FLUIDE".
dmsMessageNumber	➳		
dmsMessageMultistring	➳		
dmsMessageOwner	➳		
dmsMessageCRC	➳		
dmsMessageBeacon	➳		
dmsMessagePixelService	➳		
dmsMessageRunTimePriority	➳		
dmsMessageMsgStatus	➳		
dmsValidateMessageError	➳	2	
1203:Sign Control			
dmsControlMode	➳	4	Signification: central
dmsSWReset	➳	0	
dmsActivateMessage	➳	MessAct	String de format "messageActivationCode"
dmsMessageTimeRemaining	➳	65535	Signification: Infinite duration
dmsMsgTableSource	➳	MessID	String de format "message IDcode": mem type7, mem no255, CRC 0 (message noir affiché)
dmsMsgRequesterID	➳	127.0.0.1	
dmsMsgSourceMode	➳	8	Signification: central
dmsMemoryMgmt	➳	2	Signification: normal
dmsActivateMsgError	➳	2	Signification: None. Certains essais ont permis de valider les codes d'erreurs 7 et 8
1203:Default Message			
dmsShortPowerRecoveryMessage	➳	MessID	MessageIDCode (type7,MessNo1,CRC=0) =>NOIR
dmsLongPowerRecoveryMessage	➳	MessID	MessageIDCode (type7,MessNo1,CRC=0) =>NOIR
dmsShortpowerLossTime	➳	60	
dmsResetMessage	➳	MessID	MessageIDCode (type7,MessNo1,CRC=0) =>NOIR
dmsCommunicationLossMessage	➳	MessID	MessageIDCode (type7,MessNo1,CRC=0) =>NOIR
dmsTimeCommLoss	➳	0	
dmsPowerLossMessage	➳	MessID	MessageIDCode (type7,MessNo1,CRC=0) =>NOIR
dmsEndDurationMessage	➳	MessID	MessageIDCode (type7,MessNo1,CRC=0) =>NOIR
1203:Pixel Service			
vmsPixelServiceDuration	!		Objets NTCIP existants dans le contrôleur PMV mais la fonctionnalité n'a pu être activé
vmsPixelServiceFrequency	!		
vmsPixelServiceTime	!		

1203:Multi Error			
dmsValidateMessageError	➤	2	Signification: None
dmsMultiSyntaxError	➤	2	Signification: None
dmsMultiSyntaxErrorPosition	➤	0	
dmsMultiOtherErrorDescription	➤	0	
1203:Illumination / Brightness			
dmsIllumControl	➤	2	Signification: photocell
dmsIllumMaxPhotocellLevel	➤	16000	
dmsIllumPhotocellLevelStatus	➤	8162	
dmsIllumNumBrightLevels	➤	255	
dmsIllumBrightLevelStatus	➤	170	
dmsIllumManLevel	➤	1	
dmsIllumBrightnessValues	➤		Objet NTCIP disponible, fonctionnalité non testé
dmsIllumBrightnessValuesError	➤	2	
dmsIllumLightOutputStatus	➤	37265	
1203:Auxiliary IO			
MaxAuxDigital	➤	112	
MaxAuxAnalog	➤	24	
auxIOTable			
auxPortType			Non validé
auxPortNumber			Non validé
auxDescription			Non validé
auxResolution			Non validé
auxValue			Non validé
auxPortDirection			Non validé
1203:Sign Status			
statMultiFieldRows	➤	11	
statMultiFieldTable	➤		non accessible
statMultiFieldIndex	➤		non accessible
statMultiFieldCode	➤	1	
statMultiCurrentFieldValue	➤	03:17	
statCurrentSpeed	➤	0	
statCurrentSpeedLimit	➤	0	
watchdogFailureCount	➤	2	
dmsStatDoorOpen	➤	0	
shortErrorStatus	➤	0	
controllerErrorStatus	➤	0	

1203:Pixel Error Status			
pixelFailureTableNumRows	➳	0	
pixelFailureTable	➳		non accessible
pixelFailureDetectionType	➳	0	
pixelFailureIndex	➳		non accessible
pixelFailureXLocation	➳	0	
pixelFailureYLocation	➳	0	
pixelFailureStatus	➳	0	
pixelTestActivation	➳	2	
1203:Lamp Error			
LampFailureStuckOn	X		
LampFailureStuckOff	X		
LampTestActivation	➳	2	
1203:Fan Error			
fanfailures	➳	0	
fanTestActivation	➳	2	
1203:Power Status			
signvolts	➳	0	Aurrait probablement du donner une valeur
lowFuelThreshold	➳	0	
fuelLevel	➳	0	
engineRPM	➳	0	
lineVolts	➳	0	Aurrait probablement du donner une valeur
powerSource	➳	4	Signification: AC line
1203:Temperature Status			
tempMinCtrlCabinet	➳	0	deg C
tempMaxCtrlCabinet	➳	0	deg C
tempMinAmbiant	➳	-6	deg C
tempMaxAmbiant	➳	-6	deg C
tempMinSignHousing	➳	2	deg C
tempMaxSignHousing	➳	2	deg C
1201:Global Configuration			
globalSetIDParameter	➳	0	
globalMaxModules	➳	8	
globalModuleTable	➳		non accessible

moduleNumber	➳	1	
moduleDeviceNode	➳		valeur: 1.3.6.1.4.1.1206.4.2.3
moduleMake	➳		valeur: Daktronics Inc.
moduleModel	➳		valeur: ED-11281
moduleVersion	➳		valeur: 3
moduleType	➳		valeur: 3.2.7.21
1201:Database Management			
dBCreationTransaction			N/A
dbVerifyStatus			N/A
dbVerifyError			N/A
1201:Time Management			
globalTime	➳		valeur: 1103301500
globalDaylightSaving	➳	3	
globalLocalTimedifferential	➳		valeur: 4294963696
1201:Time Base Event Schedule			
maxTimeBaseScheduleEntries	➳	0	
timebaseScheduleTable	➳		non accessible
timebaseScheduleNumber	➳	1	
timebaseScheduleMonth	➳	0	
timebaseScheduleDay	➳	0	
timebaseScheduleDate	➳	0	
timebaseScheduleDayPlan	➳	1	
maxDayPlans	➳	20	
maxDayPlanEvents	➳	12	
timebasedayPlanTable	➳		non accessible
dayPlanNumber	➳	0	
dayPlanEventNumber	➳	0	
dayPlanHour	➳	0	
dayPlanMinute	➳	0	
dayPlanActionNumberOID	➳		valeur: 13.6.1.4.1.1206.4.2.6.3.1.0
dayPlanStatus	➳	0	
1201:Report			
maxEventLogConfigs	➳	50	
eventLogConfigTable	➳		non accessible
eventConfigID	➳		objet existant mais fonctionnalité non testé
eventConfigClass	➳		objet existant mais fonctionnalité non testé
eventConfigMode	➳		objet existant mais fonctionnalité non testé
eventConfigCompareValue	➳		objet existant mais fonctionnalité non testé

eventConfigCompareValue2	➳		objet existant mais fonctionnalité non testé
eventConfigCompareOID	➳		objet existant mais fonctionnalité non testé
eventConfigLogOID	➳		objet existant mais fonctionnalité non testé
eventConfigAction	➳		objet existant mais fonctionnalité non testé
maxEventLogSize	➳		objet existant mais fonctionnalité non testé
eventLogTable	➳		objet existant mais fonctionnalité non testé
eventLogClass	➳		objet existant mais fonctionnalité non testé
eventLogNumber	➳		objet existant mais fonctionnalité non testé
eventLogID	➳		objet existant mais fonctionnalité non testé
eventLogTime	➳		objet existant mais fonctionnalité non testé
eventLogValue	➳		objet existant mais fonctionnalité non testé
maxEventClasses	➳		objet existant mais fonctionnalité non testé
eventClassTable	➳		objet existant mais fonctionnalité non testé
eventClassNumber	➳		objet existant mais fonctionnalité non testé
eventClassLimit	➳		objet existant mais fonctionnalité non testé
eventClassClearTime	➳		objet existant mais fonctionnalité non testé
eventClassDescription	➳		objet existant mais fonctionnalité non testé
eventClassNumRowsInLog	➳		objet existant mais fonctionnalité non testé
Security			
communityNameAdmin	➳	public ou admin	Les autres objets ne sont accessible qu'en utilisant le nom de communauté "admin"
communityNameMax	➳		
communityNameTable	➳		non accessible
communityNameIndex	➳		
communityNameUser	➳		
communityNameMask	➳		

Légende	➳	Indique que l'objet NTCIP est existant dans le contrôleur PMV, il est vu par le serveur de communication et que la fonctionnalité a été vérifiée
	!	Indique que l'objet NTCIP est existant dans le contrôleur PMV, il est vu par le serveur de communication mais que la fonctionnalité n'a pu être vérifié. Pour les besoins de validation du serveur de communication ceci indique que le serveur supporte l'objet cependant le manufacturier ne supporte pas la fonctionnalité ou la fonctionnalité n'a pu être testée.
	X	Indique que le serveur de communication n'arrive pas à lire cet objet du contrôleur de PMV. Pris seul ce test pourrait indiquer que le serveur de communication ou le contrôleur de PMV ne supporte pas l'objet. Cependant, en le prenant avec les essais sur le contrôleur du 2ieme manufacturier on peut conclure que l'objet est supporté par le serveur de communication, cependant il ne l'est pas par le contrôleur de PMV.

Groupe de conformité/ Objets NTCIP	Objet supporté par	Valeur de l'objet	Commentaire
1203 :Sign Configuration and Capability			
dmsSignType	➳	134	Signification: Portable VMS full matrix
dmsBeacontype	➳	2	Signification: none
1203:GUI Appearance			
dmsSignAccess	➳	8	Signification: front access
dmsSignHeight	➳	2013	mm
dmsSignWidth	➳	3350	mm
dmsHorizontalBorder	➳	25	mm
dmsVerticalBorder	➳	25	mm
dmsLegend	➳	2	Signification: Aucun
dmsSignTechnology	➳	2	Signification:LED
1203:VMS Configuration			
vmsCharacterHeightPixels	➳	0	Signification: Variable (pleine matrice)
vmsCharacterWidthPixels	➳	0	Signification: Variable (pleine matrice)
vmsSignHeightPixels	➳	30	pixels
vmsSignWidthPixels	➳	72	pixels
vmsHorizontalPitch	➳	43	mm centre à centre pixel

vmsVerticalPitch	➳	43	mm centre à centre pixel
1203:Font Definition			
numfonts	➳	20	
fontTable	➳		non accessible
fontIndex	➳		non accessible
fontNumber	➳	1	
fontName	➳	Vermac font #1	
fontHeight	➳	7	
fontCharspacing	➳	2	
fontLineSpacing	➳	1	
fontVersionID	➳	61036	
maxFontCharacters	➳	255	
characterTable	!		Mérite d'être valider avec de nouveaux essais
fontIndex	!		Mérite d'être valider avec de nouveaux essais
characterNumber	!		Mérite d'être valider avec de nouveaux essais
characterWidth	!		Mérite d'être valider avec de nouveaux essais
characterBitmap	!		Mérite d'être valider avec de nouveaux essais
1203:Multi Configuration			
defaultBackgroundColor	➳	0	Signification: Black
defaultForegroundColor	➳	9	Signification: Amber
defaultFlashOn	➳	0	dixième de secs
defaultFlashOff	➳	0	dixième de secs
defaultFont	➳	3	
defaultJustificationLine	➳	3	Signification: center
defaultJustificationPage	➳	3	Signification: middle
defaultPageOnTime	➳	17	dixième de secs
defaultPageOffTime	➳	0	dixième de secs
defaultCharacterSet	➳	2	Signification: eightbit
1203:Message Table			
dmsNumPermanentMsg	➳	109	
dmsNumChangeableMsg	➳	3	3 Messages chargés par Matrikon pour essais
dmsMaxChangeableMsg	➳		
dmsFreeChangeableMemory	➳		

dmsNumVolatileMsg	➳	1	1 Messages chargé par matrikon pour essais
dmsMaxVolatileMsg	➳	5	
dmsFreeVolatileMemory	➳		
dmsMessageTable	➳		non accessible
dmsMessageMemoryType	➳		Donne les propriétés du message sélectionné selon les indexes utilisés. dmsMessageMultistring.2.1 par exemple est le code message du message 1 de la banque mémoire permanente (type2). La valeur de cet objet ce lit: "[jp1][pt17o0][jl3]TRAVAUX[nl]M[hcC9]CANIS[hcC9]S". Ce code message affiche TRAVAUX MÉCANISÉS au panneau.
dmsMessageNumber	➳		
dmsMessageMultistring	➳		
dmsMessageOwner	➳		
dmsMessageCRC	➳		
dmsMessageBeacon	➳		
dmsMessagePixelService	➳		
dmsMessageRunTimePriority	➳		
dmsMessageMsgStatus	➳		
dmsValidateMessageError	➳	2	
1203:Sign Control			
dmsControlMode	➳	4	Signification: central
dmsSWReset	➳	0	
dmsActivateMessage	➳	MessAct	String de format "messageActivationCode"
dmsMessageTimeRemaining	➳	65535	Signification: Infinite duration
dmsMsgTableSource	➳	MessID	String de format "message IDcode": mem type7, mem no255, CRC 0 (message noir affiché)
dmsMsgRequesterID	➳		validé valeur non noté
dmsMsgSourceMode	➳	8	Signification: central
dmsMemoryMgmt	➳	2	Signification: Normal
dmsActivateMsgError	➳	2	Signification: None. Certains essais ont permis de valider les codes d'erreurs 7 et 8
1203:Default Message			
dmsShortPowerRecoveryMessage	!		Mérite d'être valider avec de nouveaux essais
dmsLongPowerRecoveryMessage	!		Mérite d'être valider avec de nouveaux essais
dmsShortpowerLossTime	!		Mérite d'être valider avec de nouveaux essais
dmsResetMessage	!		Mérite d'être valider avec de nouveaux essais
dmsCommunicationLossMessage	!		Mérite d'être valider avec de nouveaux essais
dmsTimeCommLoss	!		Mérite d'être valider avec de nouveaux essais
dmsPowerLossMessage	!		Mérite d'être valider avec de nouveaux essais
dmsEndDurationMessage	➳		
1203:Pixel Service			
vmsPixelServiceDuration	!		Objets NTCIP existants dans le contrôleur PMV mais la fonctionnalité n'a pu être activé
vmsPixelServiceFrequency	!		
vmsPixelServiceTime	!		

1203:Multi Error			Un essais avec un message trop long pour le PMV mobile a permis d'obtenir les valeurs suivantes.
dmsValidateMessageError	➳	5	Signification: syntaxMULTI
dmsMultiSyntaxError	➳	5	Signification: textTooBig
dmsMultiSyntaxErrorPosition	➳	17	
dmsMultiOtherErrorDescription	➳	0	
1203:Illumination / Brightness			
dmsIllumControl	➳	2	Signification: photocell
dmsIllumMaxPhotocellLevel	➳	100	
dmsIllumPhotocellLevelStatus	➳	4	
dmsIllumNumBrightLevels	➳	10	
dmsIllumBrightLevelStatus	➳	1	
dmsIllumManLevel	➳	6	
dmsIllumBrightnessValues	➳		Objet NTCIP disponible, fonctionnalité non testé
dmsIllumBrightnessValuesError	➳	2	
dmsIllumLightOutputStatus	➳	2621	
1203:Auxiliary IO			
MaxAuxDigital	X		
MaxAuxAnalog	X		
auxIOTable	X		
auxPortType	X		
auxPortNumber	X		
auxDescription	X		
auxResolution	X		
auxValue	X		
auxPortDirection	X		
1203:Sign Status			
statMultiFieldRows	➳		
statMultiFieldTable	➳		non accessible
statMultiFieldIndex	➳		non accessible
statMultiFieldCode	➳	1	
statMultiCurrentFieldValue	➳	11:35	
statCurrentSpeed	➳	0	
statCurrentSpeedLimit	➳	50	
watchdogFailureCount	➳	0	
dmsStatDoorOpen	➳	3	Signification: 2 portes ouvertes, n'a pu être vérifié
shortErrorStatus	➳	0	
controllerErrorStatus	➳	0	

1203:Pixel Error Status			
pixelFailureTableNumRows	X		
pixelFailureTable	X		
pixelFailureDetectionType	X		
pixelFailureIndex	X		
pixelFailureXLocation	X		
pixelFailureYLocation	X		
pixelFailureStatus	X		
pixelTestActivation	X		
1203:Lamp Error			
LampFailureStuckOn	X		
LampFailureStuckOff	X		
LampTestActivation	X		
1203:Fan Error			
fanfailures	X		
fanTestActivation	X		
1203:Power Status			
signvolts	➳	1208	millivolts
lowFuelThreshold	X		
fuelLevel	X		
engineRPM	X		
lineVolts	➳	1208	millivolts (la déf.indique que l'unité devrait être volts)
powerSource	➳	7	Signification: battery
1203:Temperature Status			
tempMinCtrlCabinet	➳	-2	deg C
tempMaxCtrlCabinet	➳	-2	deg C
tempMinAmbiant	➳	9	deg C
tempMaxAmbiant	➳	9	deg C
tempMinSignHousing	➳	12	deg C
tempMaxSignHousing	➳	12	deg C
1201:Global Configuration			
globalSetIDParameter	➳		
globalMaxModules	➳		
globalModuleTable	➳		non accessible

moduleNumber	➳	1	
moduleDeviceNode	➳	-1	
moduleMake	➳		valeur:Leslie Bienaimé P.Eng.
moduleModel	➳	DMS	
moduleVersion	➳	V1	
moduleType	➳	3	
1201:Database Management			
dBCreationTransaction			N/A
dbVerifyStatus			N/A
dbVerifyError			N/A
1201:Time Management			
globalTime	➳		Objet et fonctionnalité vérifiés, valeur non notée
globalDaylightSaving	➳		Objet et fonctionnalité vérifiés, valeur non notée
globalLocalTimedifferential	X		
1201:Time Base Event Schedule			
maxTimeBaseScheduleEntries			Objet existant mais non validé pour ce PMV
timebaseScheduleTable			Objet existant mais non validé pour ce PMV
timebaseScheduleNumber			Objet existant mais non validé pour ce PMV
timebaseScheduleMonth			Objet existant mais non validé pour ce PMV
timebaseScheduleDay			Objet existant mais non validé pour ce PMV
timebaseScheduleDate			Objet existant mais non validé pour ce PMV
timebaseScheduleDayPlan			Objet existant mais non validé pour ce PMV
maxDayPlans			Objet existant mais non validé pour ce PMV
maxDayPlanEvents			Objet existant mais non validé pour ce PMV
timebasedayPlanTable			Objet existant mais non validé pour ce PMV
dayPlanNumber			Objet existant mais non validé pour ce PMV
dayPlanEventNumber			Objet existant mais non validé pour ce PMV
dayPlanHour			Objet existant mais non validé pour ce PMV
dayPlanMinute			Objet existant mais non validé pour ce PMV
dayPlanActionNumberOID			Objet existant mais non validé pour ce PMV
dayPlanStatus			Objet existant mais non validé pour ce PMV
1201:Report			
maxEventLogConfigs	X		
eventLogConfigTable	X		
eventConfigID	X		
eventConfigClass	X		
eventConfigMode	X		
eventConfigCompareValue	X		
eventConfigCompareValue2	X		
eventConfigCompareOID	X		

eventConfigLogOID	X		
eventConfigAction	X		
maxEventLogSize	X		
eventLogTable	X		
eventLogClass	X		
eventLogNumber	X		
eventLogID	X		
eventLogTime	X		
eventLogValue	X		
maxEventClasses	X		
eventClassTable	X		
eventClassNumber	X		
eventClassLimit	X		
eventClassClearTime	X		
eventClassDescription	X		
eventClassNumRowsInLog	X		
1201: Security			
communityNameAdmin	➳	public ou admin	Les autres objets ne sont accessible qu'en utilisant le nom de communauté "admin"
communityNameMax	➳		
communityNameTable	➳		non accessible
communityNameIndex	➳		
communityNameUser	➳		
communityNameMask	➳		

APPENDIX D
RAPPORT FINAL D'ESSAIS SUR SITE

Transports Québec

SERVEUR DE COMMUNICATION OPC POUR NTCIP

MANDAT N^{O} : 5230-04-QZ02
PROJET N^{O} : 20-5200-0321

RAPPORT FINAL D'ESSAIS SUR SITE
SERVEUR DE COMMUNICATION OPC POUR NTCIP

octobre 2005

Révision 04

Tecsult ref :°05-12757

TECSULT

CE RAPPORT A ÉTÉ PRÉPARÉ POUR LE MINISTÈRE DES TRANSPORTS PAR LE PERSONNEL DE TECSULT

Préparé par :

Le 14 juin 2005

Éric Bertrand, ing., Tecsult Inc

TABLE DES MATIÈRES

1. **INTRODUCTION** 1
2. **OBJET** 1
3. **RÉFERENCE** 2
3.1 Acronymes 2
3.2 Références 3
4. **ÉTENDUE DES ESSAIS** 4
4.1 Architecture du banc d'essais 4
4.2 Les essais 5
5. **RETOUR SUR LES ESSAIS** 5
5.1 Revue par groupe de conformité NTCIP 6
6. **RECOMMANDATIONS** 9
7. **CONCLUSION** 10

Liste des tableaux

Tableau 1 - Liste des acronymes 2
Tableau 2 - Dictionnaires de données et protocoles de communications 3

Liste des figures

Figure 1 - Configurations banc d'essais 4

Annexes

ANNEXE D-1 – Tableaux des résultats des essais
ANNEXE D-2 – Recommandations

1. INTRODUCTION

Ce document a pour but de présenter les essais sur site qui ont été réalisés sur le serveur de communication OPC pour NTCIP. Les procédures d'essais sont décrites aux paragraphes qui suivent. Les essais ont pour but de valider l'interopérabilité du serveur de communication avec les contrôleurs de panneaux à messages variables utilisés par le Ministère.

Ce rapport d'essais vise à servir de document final pour l'acceptation du serveur de communication par le Ministère et à la clôture du projet. Le serveur conçu par Matrikon est livré sous format CD avec le manuel de l'usager format électronique offert en anglais seulement.

2. OBJET

Tel qu'élaboré en début de projet, le serveur OPC pour NTCIP a pour but de permettre l'interopérabilité d'une interface de contrôle unique (client OPC) avec les panneaux à messages variables de différents manufacturiers. L'étendue des travaux consiste uniquement à la conception du serveur de communication et ne comprend pas le développement d'une interface de contrôle unique mais permettra celui-ci.

Afin de minimiser l'absence d'uniformisation dans l'implémentation des standards NCTIP par les différents manufacturiers, une référence a été établie dans le cadre du projet de conception du serveur de communication OPC pour NTCIP. La référence adoptée par le Ministère est le logiciel «NTCIP tester» de la firme Intelligent Devices Inc.

En résumé, le mandat consiste donc en la conception d'un serveur de communication OPC pour NTCIP supportant certains profils de communication et un ensemble d'objets NTCIP. Ceux-ci sont tels qu'il l'est défini dans le document de conception «OPC Server for NTCIP, Specific Requirements 1.4» et dont la réalisation doit respecter la référence donnée, soit le logiciel «NTCIP tester».

Comme il a été établi en cours de mandat, une attestation d'une tierce partie certifiant la conformité à NTCIP n'a pas été exigée. Il a été évalué qu'une telle attestation est difficile à obtenir pour un logiciel contrairement à un équipement. De plus, puisque les panneaux à messages avec lesquels les essais ont été réalisés ont reçu une attestation, et que Matrikon utilise aussi comme référence le logiciel de IDI, il est raisonnable d'affirmer qu'implicitement le serveur est conforme à la norme NTCIP.

3. RÉFERENCE

3.1 Acronymes

Tableau 1 - Liste des acronymes

Acronymes	Définition
ASC	Actuated traffic Signal Controller
CDPD	Cellular Digital Packet Data
CCTV	Closed Circuit Television
DCM	Data Collection and Monitoring Devices
DMS	Dynamic Message Sign
ESS	Environmental sensor stations
FSK	Frequency Shift Key
IP	Internet Protocol
ITS	Intelligent Transportation Systems
MIB	Management Information Base
NEMA	National Electrical Manufacturers Association
NTCIP	National Transportation Communications for ITS Protocol
OPC	OLE for Process Control
PICS	Protocol Implementation Conformance Statement
PMPP	Point-to-Multi-Point Protocol
PPP	Point-to-Point Protocol
SNMP	Simple Network Management Protocol
UDP	User Datagram Protocol

3.2 Références

Ce document fait référence à plusieurs standards selon leur désignation NTCIP. La liste suivante donne une référence complète, incluant la version exacte, des standards utilisés pour la conception du serveur de communication.

Tableau 2 - Dictionnaires de données et protocoles de communications

Référence NTCIP	Numéro du document	Titre
1101	NEMA TS 3.2-1996	*STMF : Simple Transportation Management Framework*
1102	OER	Octet encoding Rules
1103	STMP	Transportation Management Protocol
1201	NEMA TS 3.4-1996	*Global Object Definitions*
1203	NEMA TS 3.6-1997	*Object Definitions for Dynamic Message Signs*
2101	SP-PMPP/RS232	*Subnetwork Profile for Point to Multi-Point Protocol using RS-232*
2102	SP-PMPP/FSK	*Subnetwork Profile for Point to Multi-Point Protocol using FSK modems*
2103	SP-PPP/RS232	*Subnetwork Profile for Point-to-Point Protocol using RS 232*
2201	TP-Null	Null Transport Profile
2202	TP-Internet	*Internet Transport Profile*
2301	AP-STMF	*Simple transportation Management Framework*

4. ÉTENDUE DES ESSAIS

4.1 Architecture du banc d'essais

Les essais finaux se sont déroulés au centre de signalisation de Québec le 10 mai 2005. Pour les essais, un panneau à messages variables mobile du manufacturier Ver-Mac a été sélectionné par les représentants du Ministère. Le nouveau micrologiciel (firmware), SIGNCORE V3C, est chargé sur le PMVM du manufacturier. Les essais ont été conduits dans le deux configurations possibles avec ce PMVM, soit en direct avec un câble série directement branché au contrôleur et par lien téléphonique cellulaire.

Une deuxième séance d'essais a été réalisée sur un PMVL Daktronics à l'usine de Daktronics à Lachenaie et a été complété le 16 octobre 2005. Ces derniers essais se sont fait avec un lien direct RS232.

Ces configurations permettent donc de valider les deux profils de communication suivants, soit :

1. Un lien direct (ex : un réseau de fibre optique multi points avec le PMPP)
2. Un lien téléphonique commuté (cellulaire ou standard)

Le serveur de communication est installé sur l'ordinateur d'essais de même qu'un logiciel client OPC pour permettre de lire et d'écrire les différentes variables du panneau à messages variables. Pour le besoin de ces essais, le client OPC utilisé est le logiciel OPC Explorer de Matrikon qui permet de visualiser de façon simple, sous forme de liste rapide à consulter, toutes les variables du PMVM. Les figures suivantes illustrent l'installation du banc d'essais.

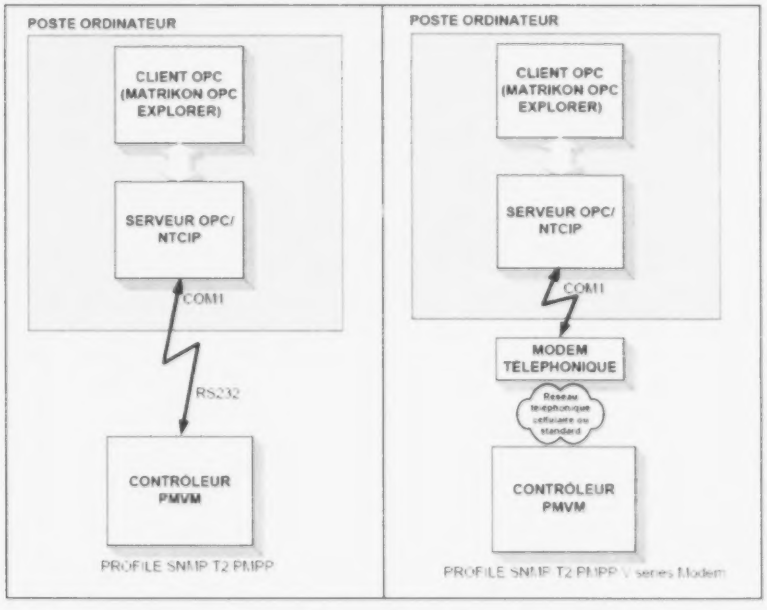

Figure 1 - Configurations banc d'essais

4.2 Les essais

De façon sommaire, le document de spécifications pour la conception du serveur de communication OPC pour NTCIP prévoit que le serveur de communication supporte tous les objets NTCIP pour les PMV. Les objets NTCIP obligatoires et optionnels supportés sont tous les objets prévus aux standards :

- NTCIP 1203 :1997 (dictionnaire d'objets pour les PMV);
- NTCIP 1201 :1996 (dictionnaire d'objets globaux).

En plus, le document de spécifications prévoit le support de certains standards et profils de communication tel qu'indiqué par les standards NTCIP suivants :

- NTCIP 2301 (SNMP);
- NTCIP 2202 (TCP et IP);
- NTCIP 2201 (T2);
- NTCIP 2101 (PMPP sur RS232);
- NTCIP 2103 (PPP sur RS232).

Les essais visent à valider le support de tous les objets NTCIP pour PMV.

En effet, le serveur expose tous les objets du MIB standard prévus par NTCIP :1203 et NTCIP 1201. Le MIB est la base de données de définition des objets qui peuvent être gérés par SNMP. Si un objet n'est pas supporté par un contrôleur du PMV, un message d'erreur est retourné au serveur lui indiquant ce fait. Les essais possibles sur un PMV particulier sont donc restreints aux objets que supporte ce contrôleur de PMV.

Le tableau d'essais est présenté en annexe. Il liste la totalité des objets obligatoires et optionnels des standards 1203 et 1201 et qui ont été testés.

Les objets NTCIP sont déclarés « supportés » et associés avec le « ☑ » dans le tableau pour un objet avec la propriété :

1. Lecture : si la valeur de cette variable peut être lue et validée pour son exactitude.
2. Écriture/Lecture : si il est possible de lire, de modifier la valeur et d'observer la fonctionnalité associée avec la nouvelle valeur.

Pour l'ensemble des objets, des essais sur les fonctionnalités ont été réalisés.

5. RETOUR SUR LES ESSAIS

Bien que le serveur puisse lire et écrire un objet du contrôleur de PMV, il faut que le PMV supporte la fonction désirée. De même, afin d'exécuter une fonction sur un PMV, l'utilisateur du serveur doit comprendre la séquence d'objets à écrire, mais aussi la façon dont le contrôleur du manufacturier s'attend à recevoir ces données. Pour la grande majorité des objets, les essais ont permis de valider que le standard était assez précis. Il a été démontré que le serveur de communication est compatible avec le contrôleur de PMV tel qu'attendu.

5.1 Revue par groupe de conformité NTCIP

5.1.1 Le groupe « Sign Configuration and Capability »

Ces variables ont une propriété de lecture seulement. Les valeurs de ces variables ont été lues et leur validité est confirmée. Le serveur est conforme.

5.1.2 Le groupe « GUI Appearance »

Ces variables ont une propriété de lecture seulement. Les valeurs de ces variables ont été lues et leur validité est confirmée. Le serveur est conforme.

5.1.3 Le groupe « VMS Configuration »

Ces variables ont une propriété de lecture seulement. Les valeurs de ces variables ont été lues et leur validité est confirmée. Le serveur est conforme.

5.1.4 Le groupe « Font Definition »

Ces variables ont les propriétés de lecture et d'écriture. Le groupe d'objets permet la création et la modification des polices de caractères affichables aux PMV. Les valeurs de ces variables ont été lues et écrites, leur validité est confirmée. Cependant, il est noté et confirmé par le manufacturier que seules les polices 15 à 20 peuvent être modifiées sur le PMVM de Vermac. Ce point fait l'objet de recommandations dans la section suivante. Pour le PMVL de Daktronics seulement les polices 1 à 3 sont modifiables. Le serveur est conforme.

5.1.5 Le groupe « Multi Configuration »

Ces variables ont les propriétés de lecture et d'écriture. Les valeurs de ces variables ont été modifiées et relues pour vérification. Ces objets sont utilisés pour définir les paramètres par défaut d'un message si ils ne sont pas spécifiés dans l'enveloppe du message. La configuration et l'affichage de quelques messages exemples permettent de démontrer le fonctionnement. Le serveur est conforme.

5.1.6 Le groupe « Message Table »

Ces variables ont les propriétés de lecture et d'écriture. Les valeurs de ces variables ont été modifiées et relues pour vérification. Ces variables donnent l'information des banques de messages de même qu'elles permettent la définition de nouveaux messages. La configuration et l'affichage de quelques messages exemples permettent de démontrer le fonctionnement. Le serveur est conforme.

5.1.7 Le groupe « Sign Control »

Ces variables ont les propriétés de lecture et d'écriture. Seule la variable « dmsMemoryMgmt » ne peut être vérifiée. La valeur de l'objet est modifiable mais un résultat n'est pas observable. Le serveur semble entièrement conforme.

5.1.8 Le groupe « Default message »

Ces variables ont les propriétés de lecture et d'écriture. Ces objets étaient non fonctionnels au moment des essais mais ont été corrigés et validés de nouveau par la suite sur le PMVM. Les objets étaient fonctionnels sur le PMVL de Daktronics. Les objets sont utilisés pour définir les messages par défaut à utiliser lorsque certains événements se produisent. Le serveur est conforme.

5.1.9 Le groupe « Pixel service »

Ces variables ont les propriétés de lecture et d'écriture. Le manufacturier confirme que la fonctionnalité de ces objets n'est pas supportée sur le PMVM. Les essais préliminaires de décembre 2004 concluent aussi que les objets sont définis dans le contrôleur PMV mais que la fonction n'est pas supportée. Les objets sont cependant supportés sur le PMVA et sont modifiable. Il est présumé que le serveur est conforme puisque les valeurs sont écrites et relues dans le contrôleur PMVL.

5.1.10 Le groupe « Multi Error »

Ces variables ont les propriétés de lecture seulement. Les valeurs de ces variables ont été lues et leur validité est confirmée par la configuration d'un message comportant une erreur. Le serveur est conforme.

5.1.11 Le groupe « Illumination / Brightness »

Ces variables ont les propriétés de lecture et d'écriture. Ils permettent de définir et contrôler les niveaux d'intensité d'affichage. Particulièrement, le mode manuel est vérifié. La courbe d'intensité est lue mais non modifiée. Le serveur est conforme.

5.1.12 Le groupe « Auxiliary IO »

Ces variables ont les propriétés de lecture et d'écriture. Le manufacturier Vermac confirme que ces objets ne sont pas supportés. Les essais sur la PMVL démontrent que les objets sont supportés par le serveur. Le serveur est conforme.

5.1.13 Le groupe « Sign Status »

Mise à part l'objet « statCurrentSpeedLimit », ces variables ont une propriété de lecture seulement. Les valeurs de ces variables ont été lues et leur validité est confirmée. Le serveur est conforme.

5.1.14 Le groupe « Pixel Error Status »

Ces variables ont les propriétés de lecture et d'écriture. Le manufacturier confirme que ces objets ne sont pas supportés sur le PMVM. Les essais sur le PMVL ont permis d'activer un test de pixels. Le serveur est conforme.

5.1.15 Le groupe « Lamp Error »

Ces variables ont les propriétés de lecture et d'écriture. Le manufacturier confirme que ces objets ne sont pas supportés sur le PMVM comme sur le PVML. Il est présumé que le serveur est conforme. Le Ministère n'utilise plus de PMV à lampes.

5.1.16 Le groupe « Fan Error »

Ces variables ont les propriétés de lecture et d'écriture. Le manufacturier confirme que ces objets ne sont pas supportés sur le PMVM comme sur le PMVL. Le serveur est conforme puisque les objets ont été vérifiés lors d'essais ponctuels sur un PMVA de Vermac.

5.1.17 Le groupe « Power status »

Ces variables ont les propriétés de lecture et d'écriture. Les variables sont partiellement vérifiées. Le manufacturier confirme que certains objets ne sont pas supportés. Le serveur est conforme.

5.1.18 Le groupe « Temperature Status »

Ces variables ont une propriété de lecture seulement. Les valeurs de ces variables ont été lues et leur validité est confirmée. Le serveur est conforme.

5.1.19 Le groupe « Global configuration »

Ces variables ont une propriété de lecture seulement. Les valeurs de ces variables ont été lues et leur validité est confirmée. Le serveur est conforme.

5.1.20 Le groupe « Time Management »

Ces variables ont les propriétés de lecture et d'écriture. Les valeurs de ces variables ont été modifiées et relues pour vérification. Le serveur est conforme.

5.1.21 Le groupe « Time Base Event Schedule »

Ces variables ont les propriétés de lecture et d'écriture. Les valeurs de ces variables ont été modifiées et relues pour vérification. La fonction est vérifiée. Le serveur est conforme.

5.1.22 Le groupe « Report »

Ces variables ont les propriétés de lecture et d'écriture. Le manufacturier confirme que ces objets ne sont pas supportés sur le PMVM. Les essais sur le PMVL de Daktronics montrent que les objets sont disponible en lecture et sont aussi modifiables. Cependant la fonction n'est pas observable. Le serveur est conforme.

5.1.23 Le groupe « Security »
Ces variables ont les propriétés de lecture et d'écriture. Les valeurs de ces variables ont été modifiées et relues pour vérification. Le serveur est conforme.

6. RECOMMANDATIONS

L'implémentation d'un serveur de communication OPC pour NTCIP permet l'uniformisation des processus de communication avec les panneaux à messages variables des différents manufacturiers. Ainsi, il permet une communication transparente avec un panneau ou l'autre. Le serveur, étant utilisé dans une architecture de contrôle d'un centre de gestion de circulation, permet l'interopérabilité de PMV de différents manufacturiers sur une même infrastructure de communication mais permet aussi, théoriquement, l'interchangeabilité des équipements.

Cependant, l'atteinte de ces objectifs est conditionnelle à certains pré requis. Les premiers pré requis sont la définition d'un dictionnaire uniforme et l'uniformisation des processus de communication pour tous les PMV. Ces deux objectifs sont atteints par la norme NTCIP et le serveur de communication.

Les normes NTCIP définissent bien le dictionnaire des objets mais laisse parfois place à interprétation au niveau de la « phraséologie ». Afin d'atteindre l'objectif final et réussir une uniformisation complète dans la gestion des panneaux à messages variables à travers une interface de contrôle unique, il importe que le Ministère standardise l'utilisation de certains objets et groupe d'objets NTCIP. Puisqu'une interface unique de contrôle doit gérer les PMV de façon transparente, on doit s'attendre à ce qu'une séquence de commandes ou l'écriture d'une variable produise exactement le même effet sur un PMV ou un autre.

Les essais ont démontré que pour la grande majorité des fonctionnalités, ce constat était observé d'un manufacturier à l'autre.

Pour permettre enfin l'intégration finale de PMV de différents manufacturiers sur un même réseau de communication, gérer par un même logiciel de contrôle, il est recommandé de préciser certains objets NTCIP au manufacturier, lors de la fourniture des PMV.

Principalement, il sera nécessaire de préciser les attentes au niveau des objets du groupe de conformité « Font Definition ». En effet, les objets de ce groupe définissent l'allure des polices de caractères, donc l'allure du message affiché sur un PMV. Un logiciel de contrôle s'attendra à retrouver les mêmes polices de caractères définies de la même façon dans les mêmes espaces mémoires du contrôleur des PMV. Cette uniformisation permet à une interface de contrôle d'offrir une prévisualisation à l'échelle d'un message à afficher. Autrement, la fonction n'est pas possible.

La recommandation est donc d'assurer dans la fourniture de PMV que six (6) polices de caractères standard du Ministère soient chargées en usine aux positions 1 à 6. Les polices 7 à 12 doivent demeurer modifiables pour donner la possibilité de créer de nouveaux jeux de caractères. Les polices 13 et suivantes peuvent être utilisées par les manufacturiers pour leurs propres besoins. De même, certains des paramètres définissant les polices doivent aussi être uniformisés. L'annexe B présente sous forme de tableau la recommandation pour la définition des variables NTCIP pour les polices de caractères.

7. CONCLUSION

En conclusion, les essais réalisés furent concluants. Il a été démontré que tous les objets NTCIP pour les PMV, soit les objets des standards NTCIP1203 et NTCIP1201, sont supportés par le serveur de communication OPC pour NTCIP. Le serveur ainsi conçu permettra le développement et l'utilisation d'une interface unique pour contrôler les PMV de différents manufacturiers. Certains points mineurs ont été relevés lors des essais finaux et ont été corrigés par Matrikon depuis.

Il est à considérer aussi que les objets propriétaires des manufacturiers ne sont pas supportés par le serveur de communication, le but étant d'uniformiser et de standardiser les mécanismes de communication avec les différents PMV peu importe le manufacturier. Si certains objets et certaines fonctionnalités devaient être retenus, il est recommandé que le Ministère les intègre à titre d'objets propriétaires ministériels que tous les manufacturiers devront supporter.

Le serveur est conçu pour les combinaisons de standards de communication suivantes:

1. SNMP-T2NULL-PMPP-RS232 DIRECT (tel que fibre optique ou direct rs232)
2. SNMP-T2NULL-PMPP-RS232 MODEM (tel standard ou cellulaire)
3. SNMP-TCPIP-PPP-RS232 DIRECT(que fibre optique ou direct rs232)
4. SNMP-TCPIP-PPP-RS232 MODEM (tel standard ou cellulaire)

À noter que les profils 1 et 2 ont largement été testés sur des PMV disponibles au Ministère. Les profils 3 et 4 n'ont pu être validés faute de contrôleurs PMV supportant la combinaison de standards. Il faut tout de même comprendre que selon toutes indications, ces profils de communications supportant TCP et IP sont souhaités puisqu'ils répondront aux besoins des réseaux IP de l'avenir prochain et que l'espérance de vie de ceux-ci est plus grande. Cependant, la firme Matrikon confirme leur fonctionnalité puisque qu'ils ont été validés au bureau avec les outils de test disponibles.

ANNEXE D-1

Tableaux des résultats des essais

	Résultats observés lors des essais définitifs Vermac SIGNCOREV3C		
Groupe de conformité/ Objets NTCIP	Objet supporté	Valeur de l'objet	Commentaire
1203 :Sign Configuration and Capability			
dmsSignType	☑	134	Signification: portable VMS full matrix
dmsBeacontype	☑	2	Signification: none
1203:GUI Appearance			
dmsSignAccess	☑	8	Signification: front access
dmsSignHeight	☑	1950	mm
dmsSignWidth	☑	3275	mm
dmsHonzontalBorder	☑	25	mm
dmsVerticalBorder	☑	25	mm
dmsLegend	☑	2	Signification: Aucun
dmsSignTechnology	☑	2	Signification LED
1203:VMS Configuration			
vmsCharacterHeightPixels		0	Signification: Variable (pleine matrice)
vmsCharacterWidthPixels	☑	0	Signification: Variable (pleine matrice)
vmsSignHeightPixels	☑	30	pixels
vmsSignWidthPixels	☑	72	pixels
vmsHonzontalPitch	☑	40	mm centre à centre pixel
vmsVerticalPitch	☑	35	mm centre à centre pixel
1203:Font Definition			
numfonts	☑	20	
fontTable			Item non accessible dans NTCIP
fontIndex			Item non accessible dans NTCIP
fontNumber		1	Seuls les polices 15 a 20 des PMVM de Ver-Mac sont modifiables
fontName		Vermac font #1	Seuls les polices 15 a 20 des PMVM de Ver-Mac sont modifiables
fontHeight		7	Seuls les polices 15 a 20 des PMVM de Ver-Mac sont modifiables
fontCharspacing		2	Seuls les polices 15 a 20 des PMVM de Ver-Mac sont modifiables
fontLineSpacing		3	Seuls les polices 15 a 20 des PMVM de Ver-Mac sont modifiables
fontVersionID	☑	56968	
maxFontCharacters	☑	255	
characterTable			Item non accessible dans NTCIP
fontIndex			Item non accessible dans NTCIP
characterNumber		61	
characterWidth		5	
characterBitmap		003e0f8000	
1203:Multi Configuration			
defaultBackgroundColor		0	Signification: Black
defaultForegroundColor		9	Signification: Amber
defaultFlashOn		0	dixième de secs
defaultFlashOff		0	dixième de secs
defaultFont		3	
defaultJustificationLine		3	Signification: center
defaultJustificationPage		3	Signification: center

defaultPageOnTime		17	dixième de secs
defaultPageOffTime		0	dixième de secs
defaultCharacterSet		2	Signification: eightbit
1203:Message Table			
dmsNumPermanentMsg	☑	109	
dmsNumChangeableMsg	☑	6	
dmsMaxChangeableMsg	☑	255	
dmsFreeChangeableMemory	☑	99600	
dmsNumVolatileMsg	☑	0	
dmsMaxVolatileMsg	☑	64	
dmsFreeVolatileMemory	☑	25600	
dmsMessageTable			Item non accessible dans NTCIP
dmsMessageMemoryType	☑	7	Après la mise au noir du panneau
dmsMessageNumber	☑	1	Après la mise au noir du panneau
dmsMessageMultistring		--	Après la mise au noir du panneau
dmsMessageOwner		Matrikon	Après la mise au noir du panneau
dmsMessageCRC		0	Après la mise au noir du panneau
dmsMessageBeacon		0	Après la mise au noir du panneau
dmsMessagePixelService		0	Après la mise au noir du panneau
dmsMessageRunTimePriority		255	Après la mise au noir du panneau
dmsMessageMsgStatus		4	Après la mise au noir du panneau
dmsValidateMessageError	☑	0	Après la mise au noir du panneau
1203:Sign Control			
dmsControlMode		4	Paramètre changé RIEN Observé sur le pmvm Vermac
dmsSWReset		1	Reset confirmé par l'activation du message "dmsResetMessage" après la commande
dmsActivateMessage		3,1,255,65535,4559	MemoryType=3, MessageNumber=1, priority=255, duration=65535, CRC=4559
dmsMessageTimeRemaining	☑	65535	
dmsMsgTableSource	☑	3,1,4559	MemoryType=3, MessageNumber=1, CRC=4559
dmsMsgRequesterID	☑	127,0,0,1	
dmsMsgSourceMode	☑	8	
dmsMemoryMgmt	?	3	Le PMV ne réagit pas à la commande
dmsActivateMsgError	☑	2	
1203:Default Message			
dmsShortPowerRecoveryMessage			Non fonctionnel au moment des essais, corrigé par la suite
dmsLongPowerRecoveryMessage			Non fonctionnel au moment des essais, corrigé par la suite
dmsShortpowerLossTime			Non fonctionnel au moment des essais, corrigé par la suite
dmsResetMessage			Non fonctionnel au moment des essais, corrigé par la suite
dmsCommunicationLossMessage			Non fonctionnel au moment des essais, corrigé par la suite
dmsTimeCommLoss			Non fonctionnel au moment des essais, corrigé par la suite
dmsPowerLossMessage			Non fonctionnel au moment des essais, corrigé par la suite
dmsEndDurationMessage			Non fonctionnel au moment des essais, corrigé par la suite
1203:Pixel Service			
vmsPixelServiceDuration			Non supporté par le PMVM (Confirmé par le manufacturier)
vmsPixelServiceFrequency			Non supporté par le PMVM (Confirmé par le manufacturier)
vmsPixelServiceTime			Non supporté par le PMVM (Confirmé par le manufacturier)
1203:Multi Error			Envoyé un message avec le caractère accentué "É" en utilisant une police qui ne définit pas les caractères accentués
dmsValidateMessageError	☑	5	
dmsMultiSyntaxError	☑	7	Character not defined
dmsMultiSyntaxErrorPosition	☑	10	position du caractère générant la faute
dmsMultiOtherErrorDescription		0	

1203:Illumination / Brightness			
dmsIllumControl		4	auto et manuel fonctionne
dmsIllumMaxPhotocellLevel	☑	65535	
dmsIllumPhotocellLevelStatus	☑	64224	
dmsIllumNumBrightLevels	☑	100	
dmsIllumBrightLevelStatus	☑	50	
dmsIllumManLevel		0-100	incrément de 10
dmsIllumBrightnessValues			courbe luminosité verifié mais non notée. Plusieurs dizaines d'octets
dmsIllumBrightnessValuesError	☑	2	
dmsIllumLightOutputStatus	☑	65535	
1203:Auxiliary IO			
MaxAuxDigital	?		Non supporté par le PMVM (Confirmé par le manufacturier)
MaxAuxAnalog	?		Non supporté par le PMVM (Confirmé par le manufacturier)
auxIOTable	?		Item non accessible dans NTCIP
auxPortType	?		Non supporté par le PMVM (Confirmé par le manufacturier)
auxPortNumber	?		Non supporté par le PMVM (Confirmé par le manufacturier)
auxDescription			Non supporté par le PMVM (Confirmé par le manufacturier)
auxResolution			Non supporté par le PMVM (Confirmé par le manufacturier)
auxValue			Non supporté par le PMVM (Confirmé par le manufacturier)
auxPortDirection			Non supporté par le PMVM (Confirmé par le manufacturier)
1203:Sign Status			
statMultiFieldRows	☑	1	
statMultiFieldTable			Item non accessible dans NTCIP
statMultiFieldIndex			Item non accessible dans NTCIP
statMultiFieldCode	☑	1	
statMultiCurrentFieldValue	☑	9 35	
statCurrentSpeed	☑	0	
statCurrentSpeedLimit		50	
watchdogFailureCount	☑	0	
dmsStatDoorOpen	☑	3,2 et 1	
shortErrorStatus	☑	0	
controllerErrorStatus	☑	0	
1203:Pixel Error Status			
pixelFailureTableNumRows	?		Non supporté par le PMVM (Confirmé par le manufacturier)
pixelFailureTable	?		Item non accessible dans NTCIP
pixelFailureDetectionType	?		Non supporté par le PMVM (Confirmé par le manufacturier)
pixelFailureIndex	?		Item non accessible dans NTCIP
pixelFailureXLocation	?		Non supporté par le PMVM (Confirmé par le manufacturier)
pixelFailureYLocation	?		Non supporté par le PMVM (Confirmé par le manufacturier)
pixelFailureStatus	?		Non supporté par le PMVM (Confirmé par le manufacturier)
pixelTestActivation			Non supporté par le PMVM (Confirmé par le manufacturier)
1203:Lamp Error			
LampFailureStuckOn	?		Non supporté par le PMVM (Confirmé par le manufacturier)
LampFailureStuckOff	?		Non supporté par le PMVM (Confirmé par le manufacturier)
LampTestActivation			Non supporté par le PMVM (Confirmé par le manufacturier)
1203:Fan Error			
fanFailures	?		Non supporté par le PMVM (Confirmé par le manufacturier)
fanTestActivation			Non supporté par le PMVM (Confirmé par le manufacturier)
1203:Power Status			
signVolts	☑	1366	
lowFuelThreshold			Non supporté par le PMVM (Confirmé par le manufacturier)
fuelLevel	?		Non supporté par le PMVM (Confirmé par le manufacturier)
engineRPM	?		Non supporté par le PMVM (Confirmé par le manufacturier)
lineVolts	☑	13	
powerSource	☑	6 et 7	solar et battery
1203:Temperature Status			
tempMinCtrlCabinet	☑	20	
tempMaxCtrlCabinet	☑	20	
tempMinAmbiant	☑	26	
tempMaxAmbiant	☑	26	

tempMinSignHousing	☑	34	
tempMaxSignHousing	☑	34	
1201:Global Configuration			
globalSetIDParameter	☑	0	
globalMaxModules	☑	2	
globalModuleTable			Item non accessible dans NTCIP
moduleNumber	☑	1	
moduleDeviceNode	☑	1 3 6 1 4 1 1206 4 2 3	
moduleMake	☑	Signalisation Ver-Mac Inc	
moduleModel	☑	Ver-Mac\SignCore™	
moduleVersion	☑	20050430-v3.92.10	
moduleType	☑	3	
1201:Time Management			
globalTime		1115736496	
globalDaylightSaving		3	Réglé l'heure du PMV qqes minutes avant le changement et observe l'heure affiché
1201:Time Base Event Schedule			
maxTimeBaseScheduleEntries	☑	3	
timebaseScheduleTable			Item non accessible dans NTCIP
timebaseScheduleNumber	☑	1	
timebaseScheduleMonth		0	
timebaseScheduleDay		0	
timebaseScheduleDate		0	
timebaseScheduleDayPlan		1	
maxDayPlans	☑	3	
maxDayPlanEvents	☑	3	
timebasedayPlanTable			Item non accessible dans NTCIP
dayPlanNumber		1	
dayPlanEventNumber		1	
dayPlanHour		0	
dayPlanMinute		0	
dayPlanActionNumberOID		1 3 6 1 4 1 1206 4 2 3 8 2 1 1 1	
dayPlanStatus		0	
1201:Report			
maxEventLogConfigs	?		Non supporté par le PMVM (Confirmé par le manufacturier)
eventLogConfigTable	?		Item non accessible dans NTCIP
eventConfigID	?		Non supporté par le PMVM (Confirmé par le manufacturier)
eventConfigClass			Non supporté par le PMVM (Confirmé par le manufacturier)
eventConfigMode			Non supporté par le PMVM (Confirmé par le manufacturier)
eventConfigCompareValue			Non supporté par le PMVM (Confirmé par le manufacturier)
eventConfigCompareValue2			Non supporté par le PMVM (Confirmé par le manufacturier)
eventConfigCompareOID			Non supporté par le PMVM (Confirmé par le manufacturier)
eventConfigLogOID			Non supporté par le PMVM (Confirmé par le manufacturier)
eventConfigAction			Non supporté par le PMVM (Confirmé par le manufacturier)
maxEventLogSize	?		Non supporté par le PMVM (Confirmé par le manufacturier)
eventLogTable	?		Item non accessible dans NTCIP
eventLogClass	?		Non supporté par le PMVM (Confirmé par le manufacturier)
eventLogNumber	?		Non supporté par le PMVM (Confirmé par le manufacturier)
eventLogID	?		Non supporté par le PMVM (Confirmé par le manufacturier)
eventLogTime	?		Non supporté par le PMVM (Confirmé par le manufacturier)
eventLogValue	?		Non supporté par le PMVM (Confirmé par le manufacturier)
maxEventClasses			Non supporté par le PMVM (Confirmé par le manufacturier)
eventClassTable	?		Item non accessible dans NTCIP
eventClassNumber	?		Non supporté par le PMVM (Confirmé par le manufacturier)
eventClassLimit			Non supporté par le PMVM (Confirmé par le manufacturier)
eventClassClearTime			Non supporté par le PMVM (Confirmé par le manufacturier)
eventClassDescription			Non supporté par le PMVM (Confirmé par le manufacturier)
eventClassNumRowsInLog	?		Non supporté par le PMVM (Confirmé par le manufacturier)

1201: Security			
communityNameAdmin		public_admin	
communityNameMax	?		Non acccessible lorsque community = "public"
communityNameTable	☑		Item non accessible dans NTCIP
communityNameIndex	☑		Item non accessible dans NTCIP
communityNameUser			Non acccessible lorsque community = "public"
communityNameMask			Non acccessible lorsque community = "public"

	Résultats observés lors des essais définitifs sur un PMVL de Daktronics		
Groupe de conformité/ Objets NTCIP	Objet supporté	Valeur de l'objet	Commentaire
1203 :Sign Configuration and Capability			
dmsSignType	☑	6	"vmsFull"
dmsBeacontype	☑	2	"none"
1203:GUI Appearance			
dmsSignAccess	☑	6	bit 1 et 2 "Walk-In Access" et "Rear Access"
dmsSignHeight	☑	3300	mm
dmsSignWidth	☑	6000	mm
dmsHorizontalBorder	☑	480	mm
dmsVerticalBorder	☑	468	mm
dmsLegend	☑	2	"NoLegend"
dmsSignTechnology	☑	2	"LED"
1203:VMS Configuration			
vmsCharacterHeightPixels	☑	0	0=full matrix
vmsCharacterWidthPixels	☑	0	0=full matrix
vmsSignHeightPixels	☑	36	pixels
vmsSignWidthPixels	☑	75	pixels
vmsHorizontalPitch	☑	66	pixels
vmsVerticalPitch	☑	66	pixels
1203:Font Definition			
numfonts	☑	10	
fontTable			
fontIndex			Indexe de la table
fontNumber.1		1	Non modifiable problème du tableau confimer par Daktronics
fontName.1		MTQ07X05_1.fnt	Non modifiable problème du tableau confimer par Daktronics
fontHeight.1		12	Non modifiable problème du tableau confimer par Daktronics
fontCharspacing.1		1	Non modifiable problème du tableau confimer par Daktronics
fontLineSpacing.1		0	Non modifiable problème du tableau confimer par Daktronics
fontVersionID.1	☑	29661	
maxFontCharacters	☑	255	
characterTable			
fontIndex			Indexe de la table
characterNumber		68	modifiable pour police 1,2,3 seulement. Confirmer par Daktronics
characterWidth		5	modifiable pour police 1,2,3 seulement. Confirmer par Daktronics
characterBitmap		0001E8C6318F8000	modifiable pour police 1,2,3 seulement. Confirmer par Daktronics
1203:Multi Configuration			
defaultBackgroundColor		0	Tableau noir valeur non modifiable
defaultForegroundColor		9	LED ambre valeur non modifiable
defaultFlashOn		0	OK modifiable
defaultFlashOff		0	OK modifiable
defaultFont		1	OK modifiable
defaultJustificationLine		2	OK modifiable
defaultJustificationPage		2	OK modifiable

defaultPageOnTime		80	OK modifiable
defaultPageOffTime		0	OK modifiable
defaultCharacterSet		2	OK modifiable
1203:Message Table			
dmsNumPermanentMsg	☑	3	
dmsNumChangeableMsg	☑	1	
dmsMaxChangeableMsg	☑	200	
dmsFreeChangeableMemory	☑	398000	
dmsNumVolatileMsg	☑	0	
dmsMaxVolatileMsg	☑	20	
dmsFreeVolatileMemory	☑	40000	
dmsMessageTable			
dmsMessageMemoryType			Indexe1 de la table
dmsMessageNumber			Indexe2 de la table
dmsMessageMultiString		MINISTERE[np]TRANSPORT	OK modifiable
dmsMessageOwner		OPC	OK modifiable
dmsMessageCRC		15546	OK modifiable
dmsMessageBeacon		0	OK modifiable
dmsMessagePixelService		0	OK modifiable
dmsMessageRunTimePriority		255	OK modifiable
dmsMessageMsgStatus		4	OK modifiable
dmsValidateMessageError	☑	2	Pas d'erreur
1203:Sign Control			
dmsControlMode		4	"Central mode"
dmsSWReset		0	Modifiable, Rien d'observable
dmsActivateMessage			Message Activation Code MessageActivationCode activatePriority=255, duration=65535, messageCRC=16135, messageMemoryType=3, messageNumber=1, sourceAddress=127.0.0.1
dmsMessageTimeRemaining	☑	65535	mins
dmsMsgTableSource	☑		MessageIDCode memoryType=3, messageNumber=1, messageCRC=0x3f07
dmsMsgRequesterID	☑	127.0.0.1	
dmsMsgSourceMode	☑	8	"central"
dmsMemoryMgmt	☑	2	"normal"
dmsActivateMsgError	☑	2	"none"
1203:Default Message			
dmsShortPowerRecoveryMessage		type=7,number=1,crc=0	OK, Modifiable
dmsLongPowerRecoveryMessage		type=7,number=1,crc=0	OK, Modifiable
dmsShortpowerLossTime		0	OK, Modifiable
dmsResetMessage		type=7,number=1,crc=0	OK, Modifiable
dmsCommunicationLossMessage		type=7,number=1,crc=0	OK, Modifiable
dmsTimeCommLoss		0	OK, Modifiable
dmsPowerLossMessage		type=7,number=1,crc=0	OK, Modifiable
dmsEndDurationMessage		type=7,number=1,crc=0	OK, Modifiable, Revient automatique à 7,255,0 après qqes secs erreur tableau
1203:Pixel Service			
vmsPixelServiceDuration		1	OK, Modifiable
vmsPixelServiceFrequency		1	OK, Modifiable
vmsPixelServiceTime		1	OK, Modifiable
1203:Multi Error			
dmsValidateMessageError	☑	0	
dmsMultiSyntaxError	☑	2	Message trop long. Le panneau coupe simplement le message et ne donne pas d'erreur
dmsMultiSyntaxErrorPosition	☑	0	
dmsMultiOtherErrorDescription		""	Pas de description

1203:Illumination / Brightness			
dmsIllumControl		2	Control en manuel possible avec 4
dmsIllumMaxPhotocellLevel	☑	255	
dmsIllumPhotocellLevelStatus	☑	34	
dmsIllumNumBrightLevels	☑	255	
dmsIllumBrightLevelStatus	☑	34	
dmsIllumManLevel		254	Control en manuel possible
dmsIllumBrightnessValues			Non disponible sur ce panneau
dmsIllumBrightnessValuesError	☑	2	
dmsIllumLightOutputStatus	☑	8974	
1203:Auxiliary IO			
MaxAuxDigital	☑	2	
MaxAuxAnalog	☑	7	
auxIOTable			
auxPortType			Indexe 1 de la table
auxPortNumber			Indexe 2 de la table
auxDescription 1		""	Suite au essais, Daktronics confirme que les indexes valables sont 2 et 3
auxResolution 1		""	Suite au essais, Daktronics confirme que les indexes valables sont 2 et 3
auxValue 1		""	Suite au essais, Daktronics confirme que les indexes valables sont 2 et 3
auxPortDirection 1		""	Suite au essais, Daktronics confirme que les indexes valables sont 2 et 3
1203:Sign Status			
statMultiFieldRows	☑	1	
statMultiFieldTable			
statMultiFieldIndex			Indexe de la table
statMultiFieldCode	☑	1 = 7	
statMultiCurrentFieldValue	☑	1= MON	Monday field displayed is 7,3 day of week
statCurrentSpeed	☑	""	Pas de radar
statCurrentSpeedLimit		""	Pas de radar
watchdogFailureCount	☑	15	
dmsStatDoorOpen	☑	1	
shortErrorStatus	☑	0	
controllerErrorStatus	☑	0	
1203:Pixel Error Status			
pixelFailureTableNumRows	☑	0	
pixelFailureTable			
pixelFailureDetectionType			Indexe de la table
pixelFailureIndex			Indexe de la table
pixelFailureXLocation 2 1	☑	""	Pas de LED en faute
pixelFailureYLocation 2 1	☑	""	Pas de LED en faute
pixelFailureStatus 2 1	☑	""	Pas de LED en faute
pixelTestActivation		2	Test activer en passant à 3
1203:Lamp Error			
LampFailureStuckOn	?	""	Non supporté par le panneau
LampFailureStuckOff	?	""	Non supporté par le panneau
LampTestActivation		2	
1203:Fan Error			
fanfailures	☑	""	Pas de ventilateur sur le panneau
fanTestActivation		""	Pas de ventilateur sur le panneau
1203:Power Status			
signvolts	☑	1326	
lowFuelThreshold		0	N/A
fuelLevel	☑	0	N/A
engineRPM	☑	0	N/A
lineVolts	☑	0	
powerSource	☑	7	"battery"
1203:Temperature Status			
tempMinCtrlCabinet	☑	37	
tempMaxCtrlCabinet	☑	37	
tempMinAmbiant	☑	4294967168	Sonde de température non existante
tempMaxAmbiant	☑	4294967168	Sonde de température non existante

tempMinSignHousing	☑	26	
tempMaxSignHousing	☑	26	
1201:Global Configuration			
globalSetIDParameter	☑	0	
globalMaxModules	☑	8	
globalModuleTable			
moduleNumber			Indexe de la table
moduleDeviceNode	☑	0 0	
moduleMake 1	☑	Daktronics, Inc	2 = Daktronics, Inc
moduleModel 1	☑	PMVL	2 = VP1000
moduleVersion 1	☑	Viper 1.0	2 = 2.1.5.4
moduleType 1	☑	2	2 = 3
1201:Time Management			
globalTime		928253244	OK, modifiable
globalDaylightSaving		3600	OK, modifiable
1201:Time Base Event Schedule			
maxTimeBaseScheduleEntries	☑	12	
timebaseScheduleTable			
timebaseScheduleNumber			Indexe de la table
timebaseScheduleMonth		8190	OK, Modifiable
timebaseScheduleDay		127	OK, Modifiable
timebaseScheduleDate		2147483646	OK, Modifiable
timebaseScheduleDayPlan		1	OK, Modifiable
maxDayPlans	☑	12	
maxDayPlanEvents	☑	12	
timebasedayPlanTable			
dayPlanNumber			Indexe1 de la table
dayPlanEventNumber			Indexe2 de la table
dayPlanHour		9	OK, Modifiable
dayPlanMinute		30	OK, Modifiable
dayPlanActionNumberOID		1 3 6 1 4 1 1206 4 2 6 3 1 0	OK, Modifiable
dayPlanStatus	☑	0	
dmsActionMsgCode 1		type=3,number=1,crc=9844	OK, Modifiable
1201:Report			
maxEventLogConfigs	☑	50	50 "event" configurables
eventLogConfigTable (50)			
eventConfigID			Indexe de la table
eventConfigClass		1	Items "REPORT" existants dans le serveur et configurable. Aucun résultat obser
eventConfigMode		2	Items "REPORT" existants dans le serveur et configurable. Aucun résultat obser
eventConfigCompareValue		0	Items "REPORT" existants dans le serveur et configurable. Aucun résultat obser
eventConfigCompareValue2		0	Items "REPORT" existants dans le serveur et configurable. Aucun résultat obser
eventConfigCompareOID		0	Items "REPORT" existants dans le serveur et configurable. Aucun résultat obser
eventConfigLogOID		0	Items "REPORT" existants dans le serveur et configurable. Aucun résultat obser
eventConfigAction		2	Items "REPORT" existants dans le serveur et configurable. Aucun résultat obser
maxEventLogSize	☑	3825	3825 "logs" disponible
eventLogTable			
eventLogClass			Indexe1 de la table
eventLogNumber			Indexe2 de la table
eventLogID	☑	""	Aucun log dans la table
eventLogTime	☑	""	Aucun log dans la table
eventLogValue	☑	""	Aucun log dans la table
maxEventClasses		15	
eventClassTable			
eventClassNumber			indexe de la table
eventClassLimit 1		4294967295	virtuellement pas de limite
eventClassClearTime 1		928256773	efface logs avant 2000
eventClassDescription 1		""	
eventClassNumRowsInLog	☑	0	

1201: Security			
communityNameAdmin			Non supporté sur le PMVL Daktronics, confirmé par le manufacturier
communityNameMax	☑		Non supporté sur le PMVL Daktronics, confirmé par le manufacturier
communityNameTable			
communityNameIndex			Indexe de la table
communityNameUser			Non supporté sur le PMVL Daktronics, confirmé par le manufacturier
communityNameMask			Non supporté sur le PMVL Daktronics, confirmé par le manufacturier

ANNEXE D-2

Recommandations

Groupe de conformité/ Objets NTCIP	Valeur de l'objet	Commentaire
numfonts	12	Minimum 12 polices de caratères disponibles
fontName 1	TypeE 7x5 fixe	type E, pour caractères de 7 X 5 pixels
fontName 2	TypeEmod 7x6 fixe	type E modifié, pour caractères de 7 X 6 pixels pour double trait
fontName 3	TypeE 9x7 fixe	type E, pour caractères de 9 X 7 pixels
fontName 4	TypeEmod 9x8 fixe	type E modifié, pour caractères de 9 X 8 pixels pour double trait
fontName 5	TypeE 7x5 mobile	type E, pour caractères de 7 X 5 pixels pour les PMVM petite lettre accentuée
fontName 6	TypeEmod 7x6 mobile	type E modifié, pour caractères de 7 X 6 pixels pour double trait petite lettre accentuée
fontName 7	Police7	Définissable par l'utilisateur
fontName 8	Police8	Définissable par l'utilisateur
fontName 9	Police9	Définissable par l'utilisateur
fontName 10	Police10	Définissable par l'utilisateur
fontName 11	Police11	Définissable par l'utilisateur
fontName 12	Police12	Définissable par l'utilisateur
fontHeight 1	12	7 pixels pour le caractère + 3 pour l'accent + 2 pour la cédille
fontHeight 2	12	7 pixels pour le caractère + 3 pour l'accent + 2 pour la cédille
fontHeight 3	14	9 pixels pour le caractère + 3 pour l'accent + 2 pour la cédille
fontHeight 4	14	9 pixels pour le caractère + 3 pour l'accent + 2 pour la cédille
fontHeight 5	9	7 pixels pour le caractère (inclus caractère accentué) + 2 pour la cédille
fontHeight 6	9	7 pixels pour le caractère (inclus caractère accentué) + 2 pour la cédille
fontCharspacing 1	1	L'espacement de caractère est défini à un (1) par défault mais est modifiable par l'utilisateur
fontCharspacing 2	1	L'espacement de caractère est défini à un (1) par défault mais est modifiable par l'utilisateur
fontCharspacing 3	1	L'espacement de caractère est défini à un (1) par défault mais est modifiable par l'utilisateur
fontCharspacing 4	1	L'espacement de caractère est défini à un (1) par défault mais est modifiable par l'utilisateur
fontCharspacing 5	1	L'espacement de caractère est défini à un (1) par défault mais est modifiable par l'utilisateur
fontCharspacing 6	1	L'espacement de caractère est défini à un (1) par défault mais est modifiable par l'utilisateur
fontLineSpacing 1	0	L'espacement de ligne est défini dans le message et est modifiable de 1 à 3 pixels
fontLineSpacing 2	0	L'espacement de ligne est défini dans le message et est modifiable de 1 à 3 pixels
fontLineSpacing 3	0	L'espacement de ligne est défini dans le message et est modifiable de 1 à 3 pixels
fontLineSpacing 4	0	L'espacement de ligne est définit dans le message et est modifiable de 1 à 3 pixels
fontLineSpacing 5	0	L'espacement de ligne est définit dans le message et est modifiable de 1 à 3 pixels
fontLineSpacing 6	0	L'espacement de ligne est définit dans le message et est modifiable de 1 à 3 pixels